پردیس علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

شناسایی نوع سوال فارسی با استفاده از روش TF-IGM برای وزن‌دهی به عبارات

نگارنده

غزاله رستمی

استاد راهنما : دکتر هدیه ساجدی

پروژه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

تیر ۱۳۹۷

چکیده

با توجه به حجم بالای داده های غیر ساختاری تولید شده توسط انسان که موجب عدم کارکرد مناسب موتور های جستجو در ارائه ی پاسخ مناسب به سوالات است نیاز به یک سیستم طبقه بندی سوال مطرح می‌شود. در یک سیستم طبقه بندی سوال هر سوال می تواند در یک ، چند و یا هیچ کلاسی قرار بگیرد . این موضوع می‌تواند در قالب یک یادگیری خودکار قرار گیرد تا بتوان به کمک آن هر سوال را به طور خودکار به دسته ای نسبت داد. هدف پروژه طراحی سیستمی برای طبقه بندی سوالات فارسی مبتنی بر روش وزن دهی TF-IGM به عنوان جایگزینی مناسب تر برای روش های سنتی TF-IDF است. این مدل یک مدل آماری جدید برای اندازه گیری قدرت تمایز میان دسته ها است. وزن دهی به عبارات مسئله ی مهمی در طبقه بندی متون است و به طور مستقیم روی دقت دسته بندی تاثیر دارد . از آنجا که روش TF-ID سنتی در دسته بندی چندان تاثیر گذار نیست روش های وزن دهی مختلفی توسط محققان پیشنهاد شده اند تاثیرگذاری این روش های پیشنهادی را با روش های دسته بندی مختلفی همچون ماشین بردار پشتیبان [۱] و k نزدیک ترین همسایه [۲] مورد بررسی قرار می دهیم . ابتدا عملیات پیش پردازش انجام می دهیم که شامل حذف کلمات کم معنی و بی معنی و علامت های نگارشی ، ریشه گیری ، بن واژه سازی و ... است . سپس با روش TF-IGM به ویژگی ها وزن تخصیص می دهیم قابل ذکر است که انتخاب ویژگی یک فاز در دسته بندی متون است و به روش های وزن دهی وابسته است ویژگی ها ی انتخاب شده باید در قالب مناسبی نمایش داده شوند در اینجا از مدل فضای برداری [۳] که یک مدل تاثیر گذار است بهره می بریم ، در این مدل هر مستند به صورت یک بردار از ویژگی ها که متناظر با آن برداری از وزن آنها داریم، نمایش داده می شود. در انتها به ارزیابی و ارائه‌ی نتایج می‌پردازیم.

[1] Support Vector Machine (SVM)
[2] K-Nearest Neighbor (KNN)
[3] Vector Space Model (VSM)

پیشگفتار

پردازش دستی اطلاعات بدون ساختار به دلیل حجم زیاد و روز افزون آنها کاری زمان بر و پر هزینه است . متن کاوی عنوانی است که برای انجام این فرآیند خودکار انتخاب شده است. از جمله کاربردهای متن کاوی دسته بندی ، خوشه بندی ، خلاصه سازی و ... است. دسته بندی متون کاربردهای زیادی دارد در این پروژه به یکی از آن ها که دسته بندی سوالات فارسی است ، اشاره می شود . طبقه بندی سوال به هیچ وجه کار بی اهمیتی نیست و نمی توان به سادگی و تنها با تکیه بر کلمات پرسشی به نتایج رضایت بخشی در زمینه ی دسته بندی سوالات رسید. ساختار های متنوع نحوی و جمله بندی سوالات دسته بندی آن ها را دشوار میکند. برای داشتن یک سیستم پاسخگویی کارا ابتدا به یک سیستم خوب برای دسته بندی سوالات نیاز داریم. در بخش دوم ، پروژه به ارائه کارهایی می پردازد که تا کنون در زمینه ی دسته بندی سوالات فارسی و انگلیسی انجام شده است. در بخش سوم به ارائه روش پیشنهادی برای وزن دهی به عبارات و همچنین به پیاده سازی سیستم طبقه بندی سوال بر پایه ی روش پیشنهادی می پردازد. در بخش چهارم به مقایسه ی روش پیشنهادی با سایر روش های وزن دهی به عبارات می پردازد . در نهایت در بخش آخر به جمع بندی و نتیجه گیری حاصل از این پروژه پرداخته می شود.

فهرست مطالب

فصل ۱

مفاهیم مقدماتی

۱.۱ آشنایی با معماری موتور جستجو

موتور جستجو نرم‌افزاری است که با گرفتن پرسشی [1] از کاربر، مستنداتی را نمایش دهد که پاسخگوی نیاز اطلاعاتی وی باشد. پرسش به معنی کلمات کلیدی است که کاربر به کمک آنها نیاز اطلاعاتی خود را بیان می‌کند. یک موتور جستجو باید مستندات را در یک ساختار مناسب جمع آوری و نگهداری کند، بتواند مشابه ترین سند به پرسش کاربر را تشخیص دهد و مستندات را به ترتیب میزان ارتباط با پرسش کاربر نمایش دهد. در اینجا موتور جستجو به ابزارهایی اطلاق می‌شود که برای استخراج اطلاعات از صفحات وب ، فایل متنی درون انباره ، رکوردهای پایگاه داده استفاده می‌شوند. پردازش اولیه ای که موتور جستجو روی پرسش انجام می‌دهد شامل مراحل زیر است :

ـ بررسی عملگرهای موجود در پرسش
ـ حذف کلمات عمومی
ـ ریشه یابی کلمات
ـ استخراج کلمات کلیدی

[1] Query

پس از این مراحل پرسش به خدمتگزار [۲] نمایه به منظور مقایسه با کلمات درون خدمتگزار ارسال می‌شود. بعد از انجام مقایسات، مستندات مرتبط شناسایی می شوند و مشخصات آنها به خدمتگزار اسناد ارسال شده تا به کاربر نمایش داده شود. موتور جستجو از سه بخش اساسی تشکیل شده است :

ـ گردآورنده اسناد
ـ نمایه ساز
ـ مدل های بازیابی اطلاعات والگوریتم دسته بندی

برخی از مشکلات زبان فارسی عبارتند از:

ـ گوناگونی معادل‌های علمی
ـ ضبط اسامی
ـ تعیین مرز کلمات (سرهم نویسی، جدانویسی، بی فاصله نویسی)
ـ انواع جمع ها
ـ صورت‌های مختلف نوشتاری
ـ استفاده از زبان محاوره در نوشتار

ریشه یابی [۳] یکی از پیچیده ترین مراحل کار در نمایه سازی متون است.اشتقاق یک واژه از ریشه اصلی موجب می‌شود تا برای ایفای نقش در جمله آماده شود. هدف ریشه یابی، زدودن الحاقات و یافتن جوهره اصلی واژه است. البته ممکن است در بعضی مواقع حذف الحاقات موجب از بین رفتن معنای واژه شود. از مزایایی که ریشه یابی دارد می توان به کاهش حجم ذخیره سازی نمایه و بهبود بازخوانی اسناد مرتبط با پرسش کاربر اشاره کرد. به عنوان مثال چند نمونه از سوال هایی که با نرم افزار پردازش زبان طبیعی دانشگاه فردوسی مشهد ریشه‌یابی کرده‌ایم را در جدول ۱.۱ مشاهده می‌کنیم

یکی از موارد مهم در نمایه ساز که نقش کلمات را از نظر میزان تاثیر آنها به عنوان کلمات کلیدی متن مشخص می‌کند وزن کلمه است. با استفاده از الگوهای مختلف وزن‌دهی، به هر واژه یا عبارت استخراج شده وزنی نسبت داده می‌شود که این وزن بیانگر میزان تاثیر کلمه در موضوع اصلی متن در مقایسه با سایر کلمات به کار رفته است.

[2]Server
[3]Stemming

سوال	سوال ریشه‌یابی شده
هیدراتهای کربن از کدام عناصر تشکیل شده اند	هیدراته کربن از کدام عنصر تشکیل کرد
به حجم پول، مجموع پول و شبه پول چه میگویند	به حجم پول، مجموع پول و شبه پول چه گفت
کوتاهترین جنگ در جهان چند روز طول کشید	کوتاه جنگ در جهان چند روز طول کشت

جدول ۱.۱: نمونه‌هایی از ریشه یابی سوالات فارسی

۲.۱ آماده‌سازی داده

آماده سازی داده به مجموعه عملیاتی که برای تولید داده‌های پالایش شده و قابل کاوش می‌شود آماده سازی داده گفته می‌شود. به بررسی دو نوع عملیات می‌پردازیم:[۶]

- استخراج داده

ابتدا داده در یک انباره داده، ذخیره می‌شود تا برای مراحل بعد مورد استفاده قرار بگیرد. این قسمت شامل دو مرحله است:

ـ جمع آوری داده

ـ ا نتخاب داده

ـ پیش‌پردازش داده

در این مرحله عملیاتی انجام می‌شود که موجب برطرف شدن مشکلات مختلف داده شده و برای انجام فرایند یادگیری آماده شود. از جمله این عملیات می‌توان به: پاک‌سازی داده، انتخاب زیر مجموعه ویژگی، فیلترینگ نمونه‌ها، نمونه‌برداری، کاهش ابعاد و... اشاره کرد. ما به معرفی دو نمونه از این عملیات می‌پردازیم :

۱.۲.۱ پاک سازی داده

یکی از مشکلات مهمی که داده می‌تواند داشته باشد این است که کیفیت آن کم باشد ابتدا مواردی که موجب کم شدن کیفیت داده می شود را معرفی می‌کنیم: نویز [۴] ،نمونه‌های پرت [۵] ،مقادیر از دست رفته [۶] ،داده‌های تکراری [۷] .

نویز : مقدار اپسیلونی اگر به داده‌ی اصلی اضافه و یا از آن کم شود موجب می‌شود از داده‌ی اصلی دور بمانیم این تغییر یا تخریب را نویز می‌گوییم.

نمونه‌های پرت : نمونه هایی که مقادیر آنها با سایر مقادیر رکوردها بسیار متفاوت است. این نمونه ها کار الگوریتم های یادگیری را دچار مشکل می کنند.

مقادیر از دست رفته : ممکن است مقادیر برخی ویژگی ها null شود راه‌هایی برای مدیریت این مقادیر از دست‌رفته وجود دارد که می توان به حذف کردن، نادیده گرفتن، تخمین زدن و جایگزین‌کردن اشاره‌کرد.

داده های تکراری : رکورد هایی هستند که بار اطلاعاتی جدیدی ندارند و اطلاعات تکراری زیادی در آنها وجود دارد. حذف داده های تکراری می‌تواند اثرات مثبت و منفی در پی داشته باشد اثر مثبت به خاطر کاهش حجم داده و اثر منفی به این دلیل که می تواند موجب از بین رفتن نظم میان داده‌ها شود.

۲.۲.۱ معرفی روش های انتخاب ویژگی

در این بخش به طور مختصر به معرفی روش های انتخاب ویژگی می‌پردازیم. از جمله عملیات برای کاهش ابعاد می‌توان به انتخاب زیرمجموعه از ویژگی‌ها اشاره نمود. در این روش ویژگی های افزونه [۸] و غیر مرتبط حذف خواهند شد. ویژگی‌های افزونه ویژگی‌هایی هستند که با توجه به ویژگی های دیگر قابل محاسبه‌اند و موجب افزایش بی دلیل فضای الگوریتم می‌شوند. ویژگی‌های نامرتبط ویژگی‌هایی هستند که هیچ ارزش اطلاعاتی برای مسأله ندارند. [۶]

---

[4] Noise
[5] Outliers
[6] Missing Values
[7] Duplicate Data
[8] Redundant Features

روش ناآگاهانه : [9] در این روش همه زیرمجموعه‌های امکان‌پذیر از ویژگی‌ها به الگوریتم داده‌کاوی اعمال خواهند شد. به طور تصادفی یک نمونه کوچک از داده‌ها انتخاب می‌شود. این روش هنگامی که تعداد ویژگی‌ها کم باشد مناسب است. در واقع در این روش الگوریتم فقط یادگیری مدل را با تمام زیرمجموعه های ممکن از ویژگی هارا بر عهده می‌گیرد و کوششی برای یادگیری هوشمندانه انجام نمی‌دهد.[۶]

روش توکار : [10] در این روش الگوریتم یادگیری مدل و انتخاب ویژگی را به صورت توأمان انجام می‌دهد.[۶]

روش فیلتری: [11] در این روش قبل از اجرای الگوریتم، انتخاب ویژگی انجام می‌شود و الگوریتم فقط یادگیری مدل را انجام می‌دهد.[۶]

روش انحصاری : [12] در این روش فقط الگوریتم وظیفه انتخاب ویژگی را دارد و یادگیری مدل را انجام نمی‌دهد.[۶]

یکی از کاربردهای مهم انتخاب زیرمجموعه ویژگی‌ها، تمرکز بر برخی ویژگی‌ها جهت یافتن ارتباط میان آنهاست.[۶]

۳.۲.۱ استخراج ویژگی از سوال

در حال حاضر برای دسته‌بندی سوالات ، انواع مختلفی از ویژگی ها وجود دارند که مورد استفاده قرار می‌گیرند از جمله‌ی این ویژگی ها که بر اساس اطلاعات زبان شناختی در نظر گرفته شده‌اند می توان سه دسته‌ی لغوی [13] ، نحوی [14] و معنایی [15] را معرفی کرد .

ویژگی‌های لغوی معمولا کلماتی هستند که در سوال ظاهر می‌شوند.
ویژگی‌های نحوی از ساختار نحوی یک سوال استخراج می‌شوند. دو نوع ویژگی نحوی رایج که در طبقه بندی سوال‌ها استفاده می‌شوند ، عبارتند از :

[9] Brute-Force Approach
[10] Embedded Approach
[11] Filter Approach
[12] Wrapper Approach
[13] Lexical
[14] Syntactic
[15] Semantic

Tagged unigrams

به هر کلمه با توجه به نقش آن در سوال یک برچسب [۱۶] اختصاص می‌دهیم . برای مثال در سوال " رییس جمهور ایران در سال گذشته چه کسی بود؟"داریم:
رییس جمهور-اسم ، ایران-اسم ، در-حرف اضافه ، سال-اسم ، گذشته-اسم ، چه کسی-کلمه پرسشی ، بود-فعل

Head words

یک سرکلمه به عنوان یک کلمه کلیدی در جمله در نظر گرفته می‌شود که حاوی اطلاعات مهمی برای شناسایی آن‌چه مورد جستجوی سوال است ، می‌باشد. شناسایی درست سر کلمه می‌تواند دقت طبقه بندی را بهبود بخشد. برای مثال : ویژگی های معنایی در مورد داده های ناقص مفید هستند . با توجه به مفهوم معنایی در سطح بالا رابطه(یا شباهت معنایی) بین کلمات شناسایی می‌شوند. سه نوع ویژگی معنایی که در دسته بندی سوال استفاده می‌شود :

Question Category(QC)

برای اینکه تشخیص دهیم یک سوال به کدام دسته تعلق دارد باید از مقایسه شباهت سر کلمه آن سوال با همه ی دسته ها استفاده کنیم کلاس با بیشترین شباهت به عنوان یک ویژگی جدید در نظر گرفته می‌شود و به بردار ویژگی اضافه می‌شود مثلا سوال کدام آهنگساز آمریکایی برای داستان غرب موسیقی نوشت ؟ سر کلمه آهنگساز است بعد از مقایسه‌ی شباهت این کلمه با همه دسته های سوال متوجه خواهیم شد بیشترین نوع شباهت با دسته‌ی بشری است .

Question Expantion(QE)

یکی دیگر از ویژگی‌های معنایی گسترش سوال است. برای هر کلمه در سوال یک وزن تعریف می کنیم که با افزایش فاصله از سر کلمه مقدار وزن کاهش می یابد.

[16] Tag

Related Words(RW)

یک دیگر از ویژگی‌های معنایی کلمات مرتبط است . کلمات در گروه‌هایی قرار می‌گیرند که هر کدام با یک نام دسته نمایان می شوند . اگر یک کلمه در یک یا چند گروه وجود داشته باشد ، مقادیر مربوط آن به بردار ویژگی اضافه می شوند . مثلا اگر هر یک از کلمات تولد ، تاریخ تولد ، روز ، دهه ، ساعت ، هفته ، ماه و سال در یک سوال باشند نام دسته (تاریخ) به بردار ویژگی اضافه می شود.

۳.۱ آشنایی با برخی روش های دسته بندی

ماشین بردار پشتیبان[۶]

استفاده از بردارهای پشتیبان خطی رویکرد جدیدی است که اخیرا مورد توجه بسیاری قرار گرفته است. این رویکرد دسته‌بندی با نظارت به این صورت عمل می‌کند که در مرحله‌ی آموزش سعی دارد مرز تصمیم گیری [۱۷] را به گونه ای انتخاب نماید که حداقل فاصله‌ی آن با هر یک از دسته‌های مورد نظر را بیشینه کند. این امر باعث می‌شود که تصمیم‌گیری ما در عمل، شرایط نویزی را به خوبی تحمل کند و پاسخ‌دهی مناسب داشته‌باشد. الگوریتم‌های مبتنی بر ماشین بردار پشتیبان الگوریتم‌هایی هستند که سعی می‌کنند یک حاشیه [۱۸] را بیشینه کنند.

الگوریتم، خطی را برای جداسازی کلاس‌های مثبت و منفی در نظر می‌گیرد که حاشیه کناری آن بیشتر باشد. لازم به ذکر است که ممکن است داده ها به گونه‌ای نباشند که بتوان با یک خط مستقیم آنها را دسته‌بندی کرد که در این شرایط برای جداسازی داده‌ها از منحنی استفاده می‌شود و با تبدیل غیرخطی، داده‌ها را به فضایی می‌بریم که بتوان آنها را با خط جدا نمود. ماشین‌های بردار پشتیبان دارای خواص زیر هستند:

ـ دسته بندی با حداکثر تعمیم
ـ رسیدن به بهینه سراسری تابع هزینه
ـ تعیین خودکار ساختار و توپولوژی بهینه برای دسته‌بند
ـ مدل کردن توابع تمایز غیرخطی با استفاده از هسته‌های غیرخطی

[17] Decision Boundary
[18] Margin

K نزدیکترین همسایه[۶]

این روش از جمله روش‌های مبتنی بر حافظه است که در دسته بندهای تاخیری [۱۹] مورد استفاده قرار می‌گیرد. دسته‌بندهای تاخیری دسته بندهایی هستند که مرحله‌ی یادگیری مدل در آن‌ها به صورت مستقل وجود ندارد یعنی مدلی یاد گرفته‌نمی‌شود. در این دسته‌بند ها کل مجموعه رکوردهای آموزشی ذخیره می‌شوند وقتی یک رکورد جدید وارد می‌شود فاصله‌اقلیدسی آن از سایر رکوردها محاسبه می‌شود و سپس دسته رکوردی که نسبت به سایر رکوردها به رکورد جدید نزدیک‌تر است به آن تخصیص داده‌می‌شود. روش K نزدیک ترین همسایه یک گروه شامل K رکورد از مجموعه رکوردهای آموزشی که نزدیک‌ترین رکوردها به رکورد آزمایشی باشند را انتخاب کرده و بر اساس برچسب مربوط به آنها در مورد دسته رکورد تصمیم‌گیری می‌کند. استفاده از این الگوریتم نیازمند تعیین سه موضوع است:

ـ باید یک مجموعه رکورد داشته باشیم.
ـ یک معیار محاسبه شباهت داشته باشیم.
ـ مقدار K نیز مشخص باشد.

برای مسائل دسته بندی دودویی بهتر است K را عددی فرد در نظر بگیریم زیرا امکان پیروز شدن یکی از دو دسته را بالا می‌برد و برای مسائل چند دسته‌ای بهتر است K را بزرگ‌تر از تعداد دسته‌ها و متفاوت با عدد تعداد دسته‌ها از لحاظ زوج یا فرد بودن در نظر بگیریم.

۴.۱ کارهای پیشین در زمینه دسته بندی سؤال

برای سیستم طبقه بندی سوال سه رویکرد پیشنهاد می شود :

۱. بر پایه قوانین اصلی
۲. یادگیری ماشین
۳. روش های ترکیبی

[19] Lazy Classifier

روش اول سعی می‌کند تا سوال‌ها را با قوانین دستی که از قبل ساخته شده‌اند متناظر کند تا بتواند نوع پاسخ سوال را شناسایی کند اما نوشتن این قوانین به صورت دستی کاری ملال آور است و باعث می‌شود سیستم نهایی بسیار خاص شود. یادگیری ماشین یک متد راحت تر به جای نوشتن قوانین برای دسته بندی سوال ها پیشنهاد می دهد که سیستم می تواند از روی داده های آموزشی یاد بگیرد و برای داده ی تست کلاس مناسب را پیش بینی کند این نوع سیستم ها می توانند خود را با شرایط جدید تطبیق دهند. سیستم های ترکیبی بسیار جدید هستند و استفاده از آن ها چندان رایج نیست. لی و روث [20] (۲۰۰۲) در یک مقاله ی علمی در زمینه ی پردازش طبیعی با استفاده از یک مجموعه ی ویژگی متنوع شامل ویژگی های نحوی و معنایی به عملکرد ۸۴.۲ درصد دست یافتند.[۴] کریشنان [21] و همکاران (۲۰۰۵) از مفهوم خبر رسان استقاده می‌کنند که یک عبارت پیوسته (یک تا سه کلمه) داخل سوال است که برای طبقه بندی دقیق مورد استفاده قرار می گیرد . آن ها یک چارچوب فراشناختی اتخاذ کردند که در آن ابتدا یک مدل توالی برای طبقه بندی خبر رسان ها ، آموزش دادند سپس ویژگی های خبر رسان پیش بینی شده را با ویژگی های کلی تر ترکیب کرده و به یک بردار بزرگ از ویژگی ها تبدیل کرده و به کمک ماشین بردار پشتیبان دسته بندی کردند و خبر رسان ها با استفاده از تجزیه سوال ورودی شناسایی شدند . این رویکرد به دقت ۸۶.۲ درصد برای دسته بندی سوال دست یافت در حالی که دقت دسته به کمک خبر رسان ها ۸۵ درصد بود.[۵]

هوانگ [22] و همکاران (۲۰۰۸) بر خلاف لی و روث که از مجموعه ویژگی ها بسیار غنی استفاده می کردند ، پیشنهاد میکنند که از مجموعه ویزگی فشرده اما موثر استفاده شود. به طور خاص آن ها ویژگی سر کلمه [23] را پیشنهاد میکنند و در مدل های ماشین بردار پشتیبان خطی و بیشترین آنتروپی [24] به ترتیب به دقت های ۸۹.۲ و ۸۹.۰ درصد برای ۵۰ کلاس ، دست یافتند.[۳]

[20] Li and Roth
[21] Krishnan
[22] Huang
[23] Head Word
[24] Maximum Entropy (ME)

اولالر ویلیامز [۲۵] (۲۰۱۰) از مجموعه ویژگی های معنایی استفاده کرده است و نشان داده است که اطلاعات معنایی به تنهایی برای تولید یک سیستم دسته بندی سوالات با کارایی بالا کافی است و سیستم نهایی اش به دقت ۸۶.۶ درصد روی دیتا ست استاندارد UIUC میرسد.

[25] Olalere Williams

فصل ۲

معرفی سیستم دسته بندی سوال بر مبنای روش وزن دهی پیشنهادی

در این قسمت مدلی جهت دسته بندی سوالات فارسی ارائه می شود سپس به ارزیابی کارایی آن پرداخته می‌شود.

۱.۲ معرفی مجموعه داده

در این گزارش از مجموعه داده (۲۰۱۶) UTQD که شامل ۱۱۷۵ سوال فارسی است، استفاده می کنیم . این سوالات به صورت دستی جمع آوری و برچسب گذاری شده اند و در قالب ۸ دسته برای طبقه بندی سوال مورد استفاده قرار می گیرند. تعداد سوالات در هر دسته در جدول ۱.۲ نشان داده شده است .

۲.۲ مدل فضای برداری

مدل فضای برداری یکی از مدل های بازیابی اطلاعات است که در سطح وسیعی به کار می‌رود. در این مدل، هر مقوله اطلاعاتی شامل متون ذخیره شده و هر تقاضای اطلاعاتی زبان طبیعی به صورت مجموعه بردارهایی از عبارات نگهداری می‌شوند. به طور نظری، این عبارات می‌توانند از واژگان کنترل شده انتخاب شوند. به خاطر وجود مشکلاتی در تهیه این واژگان،عبارات از متون استخراج می‌شوند. معمولا برای کاهش اندازه واژگان از ریشه واژه‌ها استفاده می شود. همچنین معمولا از واژه‌های بازدارنده نظیر the, of, an ,.... صرف نظر می گردد. از تمام واژه‌های موجود در مستندات ، یک مجموعه واژگان به وجود می آید.

دسته	تعداد سوالات
مخفف	۷۰
موجودیت	۱۲۹
توصیف	۱۶۰
مکان	۱۹۸
بشری	۲۵۹
عدد	۲۱۶
لیست	۳۶
آیا	۱۰۷

جدول ۱.۲: تعداد سوالات در هر دسته از مجموعه داده‌ی معرفی شده

هر مستند به صورت برداری از تمام واژگان نمایانده می شود. بعید است واژه‌هایی که فاقد بار معنایی هستند و به طور معمول در مستند یافت می‌شوند، اطلاعات مهمی ارائه دهند، بنابراین می‌توان این واژه‌ها را برای سرعت دادن به پردازش، حذف کرد. واژه‌های تکراری که می‌توان از آنها چشم پوشید فهرست واژه‌های غیرمجاز را می‌سازند. در حذف واژه‌های غیر مجاز، باید دقت زیاد به کار برده شود. برای مثال: چنانچه واژه‌های غیر مجاز در جمله: « to be or not to be » حذف شوند، این جمله غیر قابل بازیابی خواهد بود. مدل فضای برداری، شیوه ای است برای نمایش مستندات از طریق واژه‌های موجود در آنها. این مدل، یک تکنیک استاندارد در بازیابی اطلاعات است. بر اساس مدل فضای برداری، می‌توان تصمیم گرفت که کدام مستندات شبیه به یکدیگر و یا به کلید واژه‌های جستجو شبیه هستند. هم چنین برای بسیاری از روش های پردازش متن ، نیاز به نمایش عددی کلمات و متون داریم تا بتوانیم از انواع روش های عددی حوزه یادگیری ماشین مانند اکثر الگوریتم های دسته بندی روی لغات و اسناد استفاده کنیم. در این جا نقش مهم این نحوه نمایش آشکار می شود.
به طور کلی، می‌توان مزیت‌های اصلی مدل فضایی برداری را چنین بیان نمود:

۱. طرح وزن دهی به اصطلاح در این مدل، عملکرد بازیابی را بهبود می بخشد.
۲. استراتژی تطبیق جزئی این مدل، بازیابی مستنداتی را مجاز می شمارد که به شرایط جستجو نزدیک هستند.
۳. فرمول رتبه بندی کسینوسی آن، مستندات را بر طبق درجه تشابهی که به موضوع جستجو دارند، مرتب می‌کند.

۱.۲.۲ کیف لغات

فرض کنید فرهنگ لغتی داریم با N کلمه و لغت که به ترتیب الفبایی مرتب شده اند و هر لغت یک مکان مشخص در این فرهنگ لغت دارد. حال برای نمایش هر کلمه، برداری در نظر میگیریم با طول N که هر خانه آن، متناظر با یک لغت در فرهنگ لغت ماست که برای راحتی کار فرض می کنیم شماره آن خانه بردار، همان اندیس لغت مربوطه در این فرهنگ لغت خواهد بود . با این پیش فرض، برای هر لغت ما یک بردار به طول N داریم که همه خانه های آن بجز خانه متناظر با آن لغت صفر خواهد بود. در خود ستون متناظر با لغت عدد یک ذخیره خواهد شد. با این رهیافت، هر متن یا سند را هم می توان با یک بردار نشان داد که به ازای هر کلمه و لغتی که در آن به کار رفته است ، ستون مربوط از این بردار برابر تعداد تکرار آن لغت خواهد بود و تمام ستون های دیگر که نمایانگر لغاتی از فرهنگ لغت هستند که در این متن به کار نرفته اند، برابر صفر خواهد بود .

به این روش نمایش متون ، کیف لغات [۱] می گوییم که بیانگر این است که برای هر لغت در کیف یا بردار ما، مکانی درنظر گرفته شده است . با این روش ما دو بردار عددی داریم که حال می توانیم از این دو در الگوریتم های عددی خود استفاده کنیم. با وجود سادگی این روش ، اما معایب بزرگی بر آن وارد است. مثلاً اگر فرهنگ لغت ما صد هزار لغت داشته باشد ، به ازای هر متن ما باید برداری صد هزارتایی ذخیره کنیم که هم نیاز به فضای ذخیره سازی زیادی خواهیم داشت و هم پیچیدگی الگوریتم ها و زمان اجرای آنها را بسیار بالا می برد. از طرف دیگر در این نحوه مدل‌سازی فقط کلمات و تکرار آنها برای ما مهم بوده است و ترتیب کلمات یا زمینه متن (اقتصادی ، علمی ، سیاسی و...) تاثیری در مدل ما نخواهد داشت .

Word2Vec ۲.۲.۲

روشی دیگر که توسط گوگل در سال ۲۰۱۳ پیشنهاد شده است و روشی بسیار کارآمد و مناسب برای نمایش لغات و متون و پردازش آنها است روش Word2Vec است که هدف از این بخش آشنایی اولیه با این روش قدرتمند، نمایش برداری کلمات است که می تواند در بسیاری از کاربردهای نوین پردازش متن مانند سنجش احساسات، جستجوی متون مشابه یا پیشنهاد اخبار یا کالای مشابه استفاده شود.

[1] Bag of words

در این روش به کمک شبکه عصبی یک بردار با اندازه کوچک و ثابت برای نمایش تمام لغات و متون در نظرگرفته شده و با اعداد مناسب در فاز آموزش مدل [۲] برای هر لغت این بردار محاسبه می شود. در این بردار هر ستون ، نمایشگر کلمه یا ویژگی خاصی نیست و فقط یک عدد را نمایش می دهد. برای افزایش دقت این روش، مجموعه داده اولیه که برای آموزش مدل مورد نیاز است، باید حدود چند میلیارد لغت را که درون چندین میلیون سند یا متن به کار رفته اند، دربرگیرد.
بعد از ایجاد بردارهای مرتبط با هر لغت، برای نمایش برداری هر متن یا خبر ، می توان بردار تک تک کلمات به کار رفته در آن‌را یافته و میانگین اعداد هر ستون را به دست آورد که نتیجه آن یک بردار برای هر متن یا سند خواهد بود. سرعت این آموزش بسیار بالاست و در عرض چند ساعت و یا چند دقیقه (بسته به این که از کدام یک از دو الگوریتم آموزش آن استفاده کنیم) می توان حجم عظیمی از داده ها را به این الگوریتم داد و بردارهای لغات را ایجاد کرد. به طور مختصر ، این الگوریتم برای ساخت بردارهای کلمات از یکی از دو روش Skip-gram و continuous bag-of-words (CBOW) استفاده می کند. این دو روش که هر دو یک شبکه عصبی ساده هستند که بدون وجود لایه پنهانی که در اغلب روشهای شبکه عصبی وجود دارد، به کمک چند قانون ساده، بردارهای مورد نیاز را تولید می‌کنند.
در روش کیف لغات پیوسته (CBOW) ، ابتدا به ازای هر لغت یک بردار با طول مشخص و با اعداد تصادفی (بین صفر و یک) تولید می شود. سپس به ازای هر کلمه از یک سند یا متن، تعدادی مشخص از کلمات بعد و قبل آنرا به شبکه عصبی می دهیم (به غیر از خود لغت فعلی) و با عملیات ساده ریاضی، بردار لغت فعلی را تولید می کنیم (یا به عبارتی از روی کلمات قبل و بعد یک لغت، آنرا حدس می زنیم) که این اعداد با مقادیر قبلی بردار لغت جایگزین می شوند. زمانی که این کار بر روی تمام لغات در تمام متون انجام گیرد، بردارهای نهایی لغات همان بردارهای مطلوب ما هستند. روش Skip-gram برعکس این روش کار می کند به این صورت که بر اساس یک لغت داده شده ، می خواهد چند لغت قبل و بعد آنرا تشخیص دهد و با تغییر مداوم اعداد بردارهای لغات، نهایتا به یک وضعیت باثبات می رسد که همان بردارهای مورد بحث ماست.

[2] Training model

از لحاظ الگوریتمی این دو روش شبیه هم هستند با این تفاوت که CBOW لغات هدف را از روی لغات متن ورودی پیش‌بینی می‌کند ولی Skip-gram به صورت برعکس از روی لغات مرجوعه هدف، لغات ورودی را پیش‌بینی می‌کند. برعکس کردن این چرخه دلخواه به نظر می‌رسد ولی از لحاظ آماری CBOW تأثیر نرمی بر روی همه اطلاعات توزیعی دارد (با رفتاری شبیه به یک مشاهده بر روی کل متن) و درکل این روش می‌تواند روشی مفید برای استفاده در مجموعه دادگان کوچک‌تر باشد. اما Skip-gram با هر زوج محتوا-هدف به‌صورت یک مشاهده جدید رفتار می‌کند و در مجموعه دادگان بزرگ‌تر بهتر جواب می‌دهد.

۳.۲ معرفی روش پیشنهادی وزن دهی به ویژگی ها

وزن‌دهی به عبارت [۳] ها مسئله مهمی در دسته بندی متون و تأثیرگذار روی دقت دسته بندی است. از آنجا که روش TF-IDF [۴] سنتی روش چندان مؤثری نبوده است محققان روش های مختلفی برای وزن‌دهی به عبارت‌ها ارائه داده‌اند. در این بخش به مقایسه‌ی چند روش وزن‌دهی مختلف و معرفی روش TF-IGM [۵] می‌پردازیم. TF-IGM شامل یک مدل آماری است تا قدرت تمایز کلاس‌ها برای یک عبارت را به دقت اندازه بگیرد. در دسته بندی ، مستندات در مدلی به نام مدل فضای برداری نمایش داده می‌شوند در این مدل هر مستند به صورت برداری عددی نمایش داده می‌شود که شامل وزن بسیاری از عباراتی که از متن استخراج کردیم ، هستند.[۱]

بنابراین چگونگی وزن‌دهی به عبارات با یک روش مناسب مسئله‌ای اساسی در دسته بندی متون است و به طور مستقیم در دقت اثرگذار است. وزن هر عبارت بسیار وابسته به این است که یک عبارت چه میزان قدرت در تمایز دسته ها دارد. TF-IDF از این جهت که برچسب تعیین شده در مجموعه داده‌ی آموزش را مورد بررسی قرار نمی‌دهد ، روش چندان مؤثری نیست. دیبل و سباستیانی [۶] (۲۰۰۳) وزن‌دهی با نظارت [۷] را پیشنهاد دادند که دارای سه نسخه است : TF-CHI,TF-IG,TF-GR که به ترتیب با جایگزین کردن عامل IDF در TF-IDF با توابع انتخاب ویژگی information-gain,gain-ratio وχ^2 بدست می آیند .

[3] Term
[4] Term Fequancy-Inverse Document Fequancy
[5] Term Fequancy-Inverse Gravity moment
[6] Debole and Sebastiani
[7] Supervised Term weighting (STW)

اگرچه بعدها دریافتند که روش های وزن‌دهی با نظارت همواره برتر از TF-IDF نیستند، با این حال نتیجه بخش تر از آن به نظر می‌رسند. مسئله‌ی وزن‌دهی و مخصوصا وزن‌دهی با نظارت بسیار مورد توجه محققان قرار گرفت و شما های مختلفی از وزن‌دهی با نظارت توسط آن‌ها پیشنهاد شده است . شما های وزن‌دهی سنتی مانند TF و TF-IDF دودویی هستند اینکه به یک عبارت وزن بدهیم نتیجه بخش تر از این است که بر مبنای حضور یا غیابش در مستند مقدار های ۰ و ۱ قرار بدهیم چرا که بعضی از کلمات در مستند پرتکرار و بعضی کمیاب هستند و ما باید به کلمات پرتکرار وزن بیشتری بدهیم اما در این صورت و با روش TF ممکن است به عبارت هایی که پرتکرار هستند اما قدرت تفکیک بالایی ندارند، وزن زیادی بدهیم که برای رفع این نقص از فاکتور IDF استفاده می‌کنیم که در نهایت یعنی با روش TF-IDF به عبارتی با TF بالا و DF پایین وزن بیشتری تخصیص میدهیم.

فرمول محاسبه ی وزن عبارت t_k با روش TF-IDF به صورت زیر است(سباستیانی ۲۰۰۲) :

$$w(t_k) = tf_k.log(\frac{N}{df_k}) \tag{۱.۲}$$

$$w(t_k) = tf_k.log(\frac{N}{df_k} + 1) \tag{۲.۲}$$

$$w(t_k) = tf_k.log(\frac{N}{df_k}) + 1 \tag{۳.۲}$$

که tf_k ، یعنی تعداد تکرار عبارت t_k در مستند و df_k ، یعنی تعداد مستنداتی که شامل عبارت t_k هستند . به خوبی می دانیم که دسته بندی متون یک روش یادگیری ماشین با نظارت است که به مجموعه ای از متون با دسته‌ی مشخص برای آموزش مدل یا دسته بند نیاز دارد . اما TF-IDF در وزن دهی به عبارات از اطلاعات مربوط به برچسب مستندات داده‌ی آموزش استفاده نمی‌کند بنابراین وزن محاسبه شده یک عبارت به طور کامل بازتاب دهنده‌ی اهمیت آن در دسته بندی متون نیست . پس روش های وزن دهی با نظارت پیشنهاد شدند

که اغلب با جایگزینی فاکتور عمومی وزن‌دهی با یکی از تابع های انتخاب ویژگی بدست می آیند. بعد ها محققان شما های وزن‌دهی با نظارت اصلاح شده را پیشنهاد کردند به عنوان مثال میتوان به روش TF-RF [۸] که توسط لن [۹] و همکارانش (۲۰۰۹) و هم چنین روش TF-Prob [۱۰] که توسط لیو [۱۱] و همکارانش (۲۰۰۹) پیشنهاد شدند، اشاره کرد . روش TF-RF از توزیع مستندات و تعداد نمونه های کلاس مثبت و منفی در محاسبه‌ی وزن عبارات استفاده میکند. و فرمول آن به صورت زیر است:

$$w_{t_k} = tf_{t_k}.log(2 + \frac{a}{max(1, b)}) \qquad (۴.۲)$$

a تعداد مستندات در کلاس مثبت که شامل عبارت t_k هستند و b تعداد مستنداتی که در کلاس منفی شامل ترم t_k هستند را نشان می‌دهد. و فرمول روش وزن‌دهی TF-Prob هم به شکل زیر است :

$$w_{t_k} = tf_{t_k}.log(1 + \frac{a}{b}.\frac{a}{c}) \qquad (۵.۲)$$

c تعداد مستندات در کلاس مثبت که شامل عبارت t_k نیستند.

دسته ی دیگری از شماهای وزن‌دهی بر پایه‌ی ICF [۱۲] نیز وجود دارند . برای مطالعه‌ی دقیق و جامع این شما ها می توانید به مقاله ی فلان رجوع کنید . TF-IGM روش وزن‌دهی استفاده شده در این پروژه است که در گروه شماهای وزن‌دهی معنایی قرار می‌گیرد. در این روش ، وزنی که به هر عبارت تخصیص می دهیم براساس شباهت معنایی آن عبارت با یک دسته خاص است و هر دسته با یک سری کلید واژه بیان می‌شود.

[8] Term Fequancy-Relevance Fequancy
[9] Lan
[10] Term Fequancy-Probablity
[11] Liu
[12] Inverse Class Frequancy

IGM مدلی برای اندازه گیری قدرت تمایز دسته برای هر عبارت است . وزن هر عبارت با قدرت تمایز دسته آن عبارت بسیار مرتبط است برای تشخیص این قدرت تمایز به طور ابتدایی میتوان به نحوه ی توزیع آن عبارت در دسته ها توجه کرد .
به طور کلی هرچه توزیع یک عبارت در دسته ها یکنواخت تر باشد قدرت تمایز کمتری دارد . اما عبارتی که تنها در کلاس خاصی یا تعداد کمی از کلاس ها می‌آید قدرت تمایز بیشتری دارد و به چنین عباراتی باید وزن بیشتری تخصیص داد.
اگر یک عبارت تمرکز بیشتری روی توزیع بین کلاسی داشته باشد قدرت تفکیک بیشتری هم دارد . بنابراین میتوان یک عبارت را مطابق با تمرکز توزیع بین کلاسی یا غیریکنواختی آن وزن‌دهی کرد. از IGM برای اندازه گیری غیریکنواختی یا سطح تمرکز توزیع بین کلاسی یک عبارت، که بیان گر قدرت تفکیک دسته آن عبارت است ، استفاده میکنیم. برای محاسبه ی تمرکز توزیع بین کلاسی ترم t_k ابتدا فرکانس هایش در دسته های مختلف را به صورت نزولی مرتب می کنیم :
f_{kr} $(\mathrm{r}=1,2,...,\mathrm{m})$ که $f_{k1} \geq f_{k2} \geq ... \geq f_{km}$ فرکانس t_k در دسته r ام بعد از مرتب کردن است و m تعداد دسته‌هاست. هر چه تعداد دسته هایی که وقوع یک عبارت در آن‌ها متمرکز کمتر باشد مرکز گرانی [۱۳] به سر سمت چپ نزدیک‌تر است.به خصوص زمانی که یک عبارت تنها در یک دسته اتفاق می‌افتد، مرکز گرانی در مکان $\mathrm{r}=1$ است. و اگر توزیع بین دسته‌ای یکنواخت باشد یعنی $f_{k1}=f_{k2}=\cdots=f_{km}$ باشد مرکز گرانی در مکان $\mathrm{m}/2$ قرار می‌گیرد. بنابراین موقعیت مکانی مرکز گرانی بیانگر تمرکز توزیع بین دسته‌ای است. برای کل فرکانس وقوع یک عبارت در مجموعه داده هرچه تمرکز توزیع بین دسته ای بیشتر باشد فاصله ی مرکز گرانی از مبدأ کمتر خواهد شد. IGM برای اندازه گیری تمرکز توزیع بین کلاسی پیشنهاد شده است و به صورت زیر تعریف می‌شود :

$$igm(t_k) = \frac{f_{k1}}{\sum_{r=1}^{m} f_{kr}.r} \qquad (۶.۲)$$

[13] Gravity Center

فرمول (۴.۲)را میتوان به صورت زیر هم نوشت :

$$igm(t_k) = \frac{1}{\sum_{r=1}^{m} \frac{f_{kr}}{max(f_{ki})}.r} \qquad (۷.۲)$$

و فرم نرمال شده‌ی IGM به صورت زیر است :

$$nigm(t_k) = \frac{(1+m).m.igm(t_k) - 2}{(1+m).m - 2} \qquad (۸.۲)$$

بازه‌ی مقدار IGM از $\frac{2}{((1+m).m)}$ تا ۱ است و زمانی که توزیع یکنواخت باشد حداقل است و وقتی که $f_{k1}>0$ و $f_{k2}=\cdots=f_{km}=0$ باشد آنگاه حداکثر است. حال به جای فاکتور IDF سنتی در وزن‌دهی به عبارات یک فاکتور عمومی جدید بر پایه IGM به‌صورت زیر تعریف می‌شود :

$$w_g(t_k) = 1 + \lambda.igm(t_k) \qquad (۹.۲)$$

λ یک ضریب قابل تنظیم بین ۵.۰ تا ۹.۰ است. دو شمای وزن دهی به عبارت TF-IGM و RTF-IGM با فرمول های زیر تعریف می شوند :

$$w(t_k, d) = tf_{kd}.(1 + \lambda.\frac{f_{k1}}{\sum_{r=1}^{m} f_{kr}.r}) \qquad (۱۰.۲)$$

$$w(t_k, d) = \sqrt{tf_{kd}}.(1 + \lambda.\frac{f_{k1}}{\sum_{r=1}^{m} f_{kr}.r}) \qquad (۱۱.۲)$$

۴.۲ مقایسه‌ی روش پیشنهادی با سایر روش های وزن‌دهی به عبارات

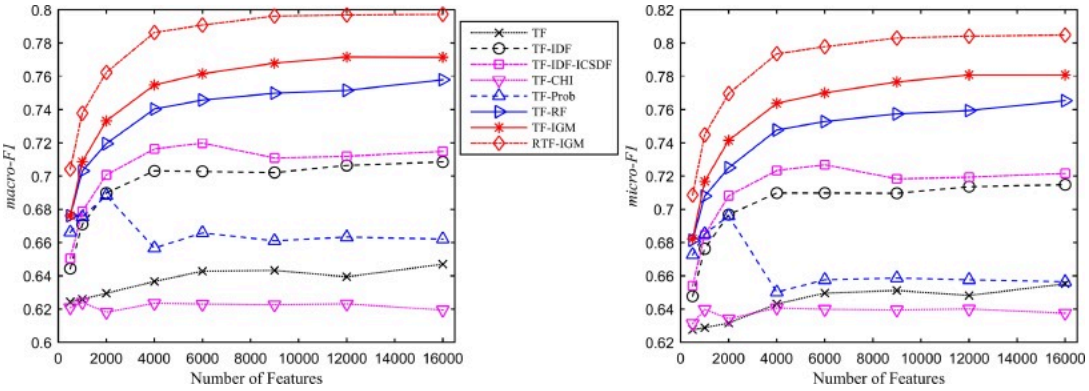
در مقاله‌ای [۱۴] که کوین چن [۱۵] و همکارانش راجع به روش وزن‌دهی TF-IGM نوشته اند، عملکرد هشت شمای وزن‌دهی مختلف در دسته بندی چند کلاسه با استفاده از دسته‌بند های SVM و kNN روی سه مجموعه داده‌ی 20 Newsgroups و Reuters-21578 و TanCorp با معیارهای ارزیابی Macro-averaged f1 و Micro-averaged f1 مورد بررسی قرار گرفته است و نتایج ارزیابی ها نشان می‌دهد که دو شمای وزن‌دهی TF-IGM و RTF-IGM عملکرد بهتری نسبت به شماهای دیگر (TF,TF-IDF,TF-CHI,TF-Prob,TF-RF) ، در دسته‌بندی چند کلاسه دارند .

با این که TF-IDF معروف ترین شمای وزن‌دهی است و ثابت شده است که TF-RF کارایی بهتری نسبت به بسیاری از شماهای دیگر دارد شماهای پیشنهادی از هر دوی آن ها هم بهتر عمل میکنند.

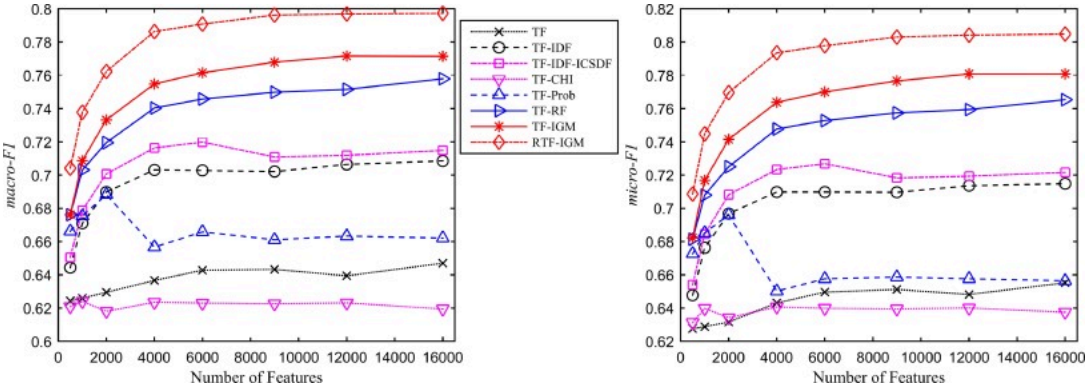
به عنوان مثال درهمین مقاله نتایج آزمایشات روی مجموعه داده‌ی 20 Newsgroups با دسته بند های SVM خطی و kNN(k=20) در شکل های ۱.۲ و ۲.۲ نشان داده شده است. هر منحنی در شکل ها بیانگر یک شمای وزن دهی است و محور عمودی نشان دهنده‌ی مقدار معیار های ارزیابی Macro-averaged f1 و Micro-averaged f1 در دسته‌بندی متون و محور افقی بیانگر تعداد واژه های متناظر است. واضح است که در دسته بندی با استفاده از SVM روی این مجموعه داده کارایی RTF-IGM و TF-IGM در هر تعداد از ویژگی ها از سایر شماهای نشان داده شده بهتر است . TF-RF و TF-ICSDF به مقدار بسیار کمی بهتر از TF-IDF هستند و TF عملکرد ضعیف‌تری نسبت به TF-IDF دارد و TF-CHI و TF-Prob عملکرد بسیار پایینی در مقایسه با بقیه دارند.

[14] Turning from TF-IDF to TF-IGM for term weighting in text classification
[15] Kewen Chen

در دسته بندی با استفاده از kNN(k=20) همانطور که در شکل ۲.۲ نشان داده شده است تفاوت کارایی میان شما ها بیشتر از تفاوتشان در حالت استفاده از دسته بند SVM است . برای هر تعداد ویژگی کارایی RTF-IGM و TF-IGM نسبت به دیگر شماها برتری دارد TF-RF و TF-ICSDF عملکرد بهتری از TF-IDF دارند. اما TF و TF-CHI و TF-Prob مقدار کمی ضعیف‌تر از TF-IDF عمل می کنند.[۱]



شکل ۱.۲: مقایسه‌ی کارایی هشت شمای وزن‌دهی مختلف با دسته‌بند SVM روی مجموعه داده‌ی 20Newsgroups



شکل ۲.۲: مقایسه‌ی کارایی هشت شمای وزن‌دهی مختلف با دسته‌بند kNN(k=20) روی مجموعه داده‌ی 20Newsgroups

فصل ۳

روش های ارزیابی و تفسیر مدل

۱.۳ معیارهای ارزیابی

در آخرین مرحله پس از پیاده سازی الگوریتم‌های دسته بندی باید با استفاده از متون آزمایشی، صحت، دقت، بازخوانی، معیار ارزیابی F مدل پیشنهادی را بدست می‌آوریم. قبل از بیان روابط سنجش دقت دسته بندی نیاز به معرفی پیش نیازهای زیر است. ابتدا به جدول ۱.۳ توجه کنید :

FP : تعدادی از داده ها که به غلط به عنوان دسته مثبت شناسایی می‌شوند.

TP : تعدادی از داده ها که به درست به عنوان دسته مثبت شناسایی می‌شوند.

FN : تعدادی از داده ها که به غلط به عنوان دسته منفی شناسایی می‌شوند.

TN : تعدادی از داده ها که به درست به عنوان دسته منفی شناسایی می‌شوند.

معیار های ارزیابی دسته

$$Recall = \frac{TP}{FN + TP} \tag{۱.۳}$$

$$Precision = \frac{TP}{FP + TP} \tag{۲.۳}$$

$$F_{measure} = \frac{2.Recall.Precision}{Recall + Precision} \tag{۳.۳}$$

	دسته مثبت	دسته منفی
دسته مثبت	TP	FP
دسته منفی	FN	TN

جدول ۱.۳: ماتریس درهم ریختگی برای مسئله‌ی دسته بندی دو دسته‌ای

$$micro-averagedF1 = \frac{2 * \sum_{i=1}^{C} |TPc_i|}{2 * \sum_{i=1}^{C} |TP_{c_i}| + \sum_{i=1}^{C} |FN_{c_i}| + \sum_{i=1}^{C} |FP_{c_i}|} \quad (۴.۳)$$

$$macro - averagedF1 = \frac{1}{C} \sum_{i=1}^{C} F1_{c_i} \quad (۵.۳)$$

$$accuracy = \frac{TP + TN}{TN + TP + FN + FP} \quad (۶.۳)$$

$$Error = \frac{FP + FN}{TN + TP + FN + FP} \quad (۷.۳)$$

۲.۳ روش‌های ارزیابی

۱.۲.۳ روش Holdout

در این روش مجموعه داده به دو بخش با نام های داده آموزشی و آزمایشی تقسیم می‌شود. مدل دسته بندی توسط داده‌ی آموزشی ساخته شده و به وسیله ی داده آزمایشی ارزیابی می‌شود. چگونگی نسبت تقسیم به تشخیص تحلیل‌گر بستگی دارد و حسن این روش سادگی و سرعت بالاست . اولین ایراد این روش آن است که محموعه داده‌ای که برای آزمایش استفاده می‌شود شانسی برای حضور در مرحله‌ی آزمایش ندارد. دومین ایراد این است که مدل بستگی به چگونگی تقسیم داده دارد اگر مجموعه داده‌ی آموزشی بزرگ در نظر گرفته شود دقت نهایی به دلیل کوچک شدن مجموعه داده‌ی آموزشی غیر قابل اعتماد خواهد بود.

۲.۲.۳ روش Random Subsampling

اگر روش Holdout را چندین بار اجرا کنیم و از نتایج حاصل میانگین گیری کنیم روش قابل اعتماد تری را برگزیده‌ایم مهمترین عیب این روش این است که در آن هیچ کنترلی بر روی تعداد دفعاتی که یک رکورد به عنوان نمونه‌ی آموزشی یا آزمایشی مورد استفاده قرار می‌گیرد وجود ندارد یعنی بعضی رکوردها ممکن است بیش از سایر رکوردها برای یادگیری یا ارزیابی به کار بروند.

۳.۲.۳ روش Cross Validation

در این روش کل مجموعه داده‌ها به k قسمت مساوی تقسیم می‌شوند. از k-1 قسمت به عنوان مجموعه داده‌های آموزشی استفاده می‌شود و براساس آن مدل ساخته می‌شود و با یک قسمت باقی مانده عملیات ارزیابی انجام می‌شود. فرآیند مزبور به تعداد k مرتبه تکرارخواهد شد، به گونه‌های که از هر کدام از k قسمت تنها یک‌بار برای ارزیابی استفاده شده و در هر مرتبه یک دقت برای مدل ساخته شده، محاسبه می‌شود. در این روش ارزیابی دقت نهایی دسته بند برابر با میانگین k دقت محاسبه شده خواهد بود .

۴.۲.۳ روش Bootstrap

در روش‌هایی که تا کنون گفته شد فرض بر آن است که انتخاب محموعه‌ی آموزشی بدون جایگذاری صورت گرفته است. حال فرض می‌کنیم که هر رکورد مجددا هم می‌تواند برای یادگیری مورد استفاده قرار گیرد. سپس رکوردهای انتخاب نشده برای ارزیابی مورد استفاده قرار می‌گیرند. این عملیات به تعداد b تکرار می‌شود احتمال انتخاب هر رکورد در مجموعه داده‌ی اولیه برابر با : $1 - (1 - \frac{1}{N})^N$ است. اگر N به اندازه کافی بزرگ انتخاب شود ، حاصل این رابطه ۰.۶۳۲ خواهد شد. به همین دلیل هر Bootstrap معادل ۰.۶۳۲ مجموعه داده‌ی اولیه خواهد شد.

فصل ۴

پیاده سازی و نتیجه گیری

۱.۴ آماده سازی و استخراج ویژگی از داده

در این بخش ابتدا عملیات پیش پردازش را روی مجموعه داده اعمال نمودیم هم‌چنین کلمات کم معنی و بی‌معنی [1] را که در یک فایل متنی ذخیره کرده بودیم، از مجموعه داده حذف نمودیم. به تعداد کلاس‌ها فایل متنی ساختیم، نام هر فایل برچسب کلاس متناظر بود . سپس در هر فایل ویژگی‌های استخراج شده از تمام سوال‌های مجموعه داده که برچسب آن‌ها نام همان فایل بود، با در نظر گرفتن تکرار قرار داده شد. همه ویژگی‌های استخراج شده در فایلی به نام Featurs قرار گرفتند. از آن‌جایی که برای ساخت بردار داشتن ویژگی تکراری کار بیهوده‌ای است با حذف ویژگی‌های تکراری این فایل، فایل جدیدی تحت عنوان Selected featurs ساختیم.

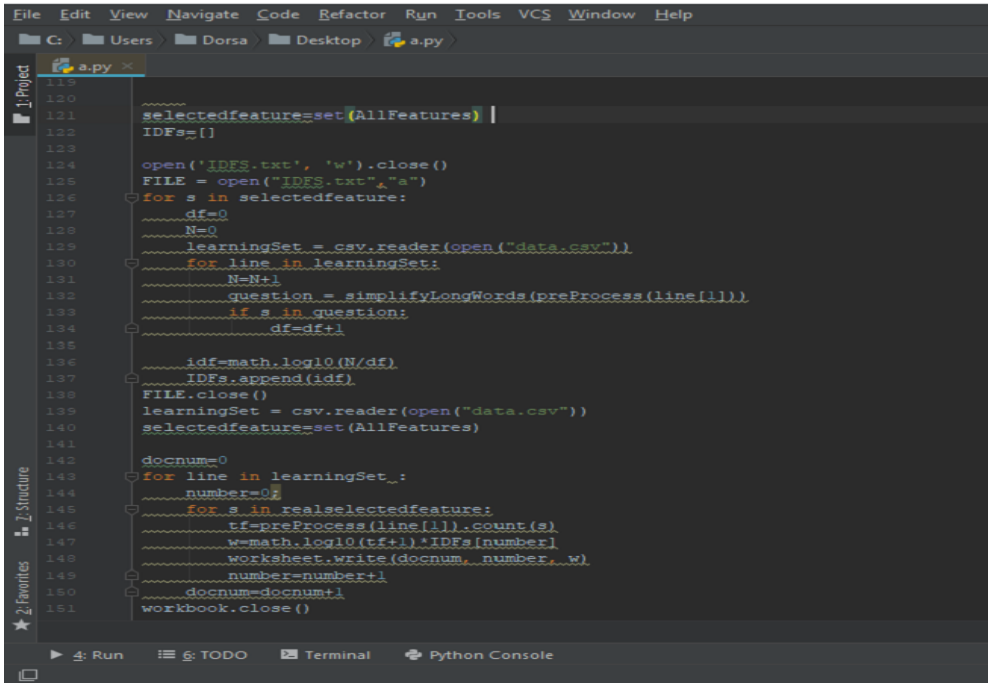
۲.۴ تولید بردار ویژگی

ماتریسی که قرار است برای دسته بندی از آن استفاده کنیم به تعداد سوالات مجموعه داده سطر و به تعداد ویژگی‌های انتخاب شده ستون دارد . اگر تنها به حضور یا عدم حضور ویژگی‌های انتخاب شده در هر سوال توجه می‌کردیم آنگاه اگر ویژگی انتخاب شده در سوال ظاهر شده بود عنصر متناظر در ماتریس را صفر و اگر ظاهر نشده بود، عنصر متناظر را یک قرار می‌دادیم.

[1] Stop words

۳.۴ وزن‌دهی با روش TF-IDF

پر واضح است که این روش، روش مناسبی نیست زیرا تکرار ویژگی در سوال را در نظر نمی‌گیرد . ما در این‌جا از روش وزن‌دهی TF-IDF استفاده می‌کنیم تا این مشکل را برطرف کرده و دقت را بهبود بخشیم. همان طور در شکل ۱.۴ مشاهده می‌کنید ابتدا df محاسبه شده است سپس به ازای هر واژه از مجموعه داده آموزشی tf محاسبه شده است و طبق فرمول TF-IDF وزن هر واژه تولید شده است.



شکل ۱.۴: کد پایتون برای وزن‌دهی به روش TF-IDF

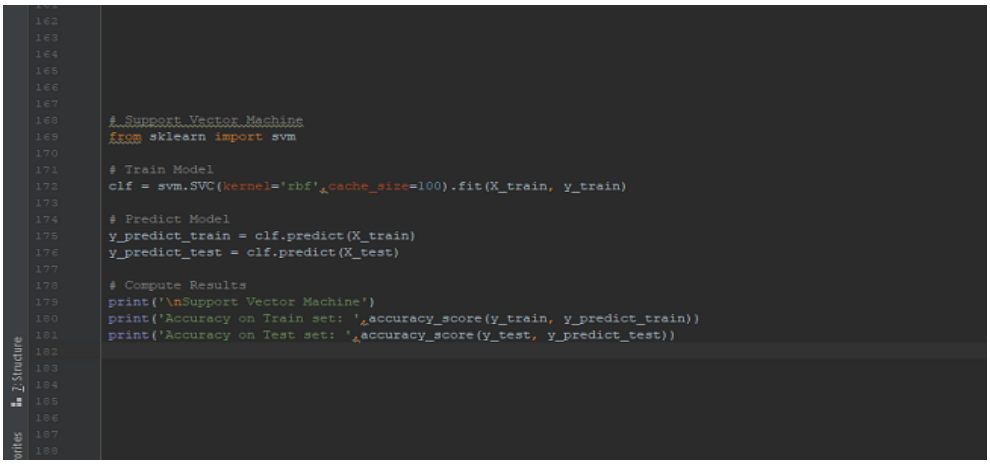
در این مرحله مجموعه داده را با نسبت ۰.۷ برای داده آموزشی و ۰.۳ برای داده آزمایشی تقسیم می‌کنیم و پس از آن ماتریس آماده شده را به دسته بند های kNN و SVM می‌دهیم. کد پایتون مورد استفاده برای این دسته‌بندی‌ها را در شکل‌های ۲.۴ و ۳.۴ مشاهده می‌کنید. دقت با دسته‌بند kNN ، ۸۲.۲ درصد ، با دسته‌بند SVM ، ۷۹.۳ درصد است.

```
129
130
131
132
133
134  # KNeighborsClassifier
135  from sklearn.neighbors import KNeighborsClassifier
136
137  # Train Model
138  clf = KNeighborsClassifier(n_neighbors=15,algorithm='ball_tree').fit(X_train, y_train)
139
140  # Predict Model
141  y_predict_train = clf.predict(X_train)
142  y_predict_test = clf.predict(X_test)
143
144  # Compute Results
145  print('\nKNeighborsClassifier')
146  print('Accuracy on Train set: ',accuracy_score(y_train, y_predict_train))
147  print('Accuracy on Test set: ',accuracy_score(y_test, y_predict_test))
148
149
150
151
```

شکل ۲.۴: کد پایتون برای دسته‌بند kNN

۴.۴ استفاده از RapidMiner برای دسته‌بندی

در ادامه با نرم افزار رپیدماینر ماتریس در هم ریختگی را تولید کردیم که آن را در شکل ۴.۴ و ۵.۴ مشاهده می‌کنید. در فصل ۳ با معیارهای ارزیابی مدل آشنا شدیم. در این ماتریس معیارهای Precision و Recall برای هر کدام از دسته‌ها نمایش داده شده است . همان طور که مشاهده می‌شود این دو معیار برای دسته‌ی DESC نسبت به سایر دسته‌ها مقدار بیشتری را به خود اختصاص داده‌اند و این به این معناست که مدل ساخته شده روی این دسته عملکرد بهتری دارد. بالا بودن معیار Precision بیانگر این است که درصد بیشتری از سوال هایی که توسط مدل کلاس آن‌ها DESC پیش بینی شده است واقعا متعلق به این کلاس هستند . هم‌چنین بالا بودن معیار Recall یعنی از میان سوال‌هایی که واقعا متعلق به کلاس DESC هستند، تعداد زیادی‌شان به درستی پیش بینی شده‌اند.



```
# Support Vector Machine
from sklearn import svm

# Train Model
clf = svm.SVC(kernel='rbf', cache_size=100).fit(X_train, y_train)

# Predict Model
y_predict_train = clf.predict(X_train)
y_predict_test = clf.predict(X_test)

# Compute Results
print('\nSupport Vector Machine')
print('Accuracy on Train set: ', accuracy_score(y_train, y_predict_train))
print('Accuracy on Test set: ', accuracy_score(y_test, y_predict_test))
```

شکل ۳.۴: کد پایتون برای دسته‌بند SVM

	true ENTY	true DESC	true ABBR	true HUM	true NUM	true LOC	class precision
pred. ENTY	335	4	2	10	65	111	63.57%
pred. DESC	1	291	0	17	0	2	93.57%
pred. ABBR	3	2	22	3	0	1	70.97%
pred. HUM	14	7	0	290	3	27	85.04%
pred. NUM	2	3	0	4	163	3	93.14%
pred. LOC	5	4	0	9	0	96	84.21%
class recall	93.06%	93.57%	91.67%	87.09%	70.56%	40.00%	

شکل ۴.۴: ماتریس در هم ریختگی داده آموزشی و دسته‌بند kNN

	true ENTY	true DESC	true ABBR	true HUM	true NUM	true LOC	class precision
pred. ENTY	53	2	0	4	0	6	81.54%
pred. DESC	0	96	0	5	0	1	94.12%
pred. ABBR	1	0	7	0	0	0	87.50%
pred. HUM	8	3	0	86	0	7	82.69%
pred. NUM	1	2	0	1	37	0	90.24%
pred. LOC	33	0	1	4	23	69	53.08%
class recall	55.21%	93.20%	87.50%	86.00%	61.67%	83.13%	

شکل ۵.۴: ماتریس در هم ریختگی داده آزمایشی و دسته‌بند kNN

منابع

[1] Kewen Chen, Zuping Zhang, and Jun Long Hao Zhang. "Turning from TF-IDF to TF-IGM for term weighting in text classification." Expert System With Applications,vol.66 (2016) (Pages 245-260)
[2] Mohammad Razzaghnoori, Hedieh Sajedi, and Iman Khani Jazani. "Question classification in Persian using word vectors and frequencies." Cognitive systems research, vol.47 (2018) (Pages 16-27)
[3] Zhiheng Huang, Marcus Thint, and Zengchang Qin . "Question classification using head words and their hypernyms. "
[4] Li, and D. Roth. "Learning Question Classifiers." The 19th international conference on computational linguistics,vol.1 (2002) (Pages 1–7)
[5] V. Krishnan, S. Das, and S. Chakrabarti. "Enhanced Answer Type Inference from Questions using Sequential Models." The conference on Human Language Technology and Empirical Methods in Natural Language Processing.(2005)
[6] صنیعی آباده محمد، محمودی سینا، طاهرپرور محدثه. " داده کاوی کاربردی." تهران، انتشارات نیاز دانش(۱۳۹۴) (صفحات ۵۲-۵۶، ۱۲۳-۱۲۶)

Abstract

With the rapid growth of textual content on the Internet, automatic text categorization is a comparatively more effective solution in information organization and knowledge management.The necessity of the existence of Question Answering (QA) systems becomes evident by considering the fact that the enormous amount of unstructured data created by humans nowadays, results in ineffectiveness of search engines to provide the exact solution for a given question. Question classification plays an important role in question answering. Features are the key to obtain an accurate question classifier. Question classifier is a system that assigns a label to each question. Feature selection, one of the basic phases in statistical-based text categorization, crucially depends on the term weighting methods In order to improve the performance of text categorization. Term weighting is a basic problem in text classification and directly affects the classification accuracy. Since the traditional TF-IDF (term frequency and inverse document frequency) is not fully effective for text classification, various alternatives have been proposed by researchers and here in this report we use the TF-IGM term weighting method in Persian question classification.

College of Science
School of Mathematics, Statistics, and Computer Science

Question classification in Persian using TF-IGM term weighting

Ghazale Rostami

Supervisor: Dr. Hedieh Sajedi

A thesis submitted to Graduate Studies Office
in partial fulfillment of the requirements for the degree of
B.Sc. in
Computer Science

2018