پردیس علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

شناسایی نوع سوال فارسی با استفاده از روش‌های تغییر یافته‌ی TF-IDF برای وزن‌دهی به عبارات

نگارنده

درسا کحقی

استاد راهنما : دکتر هدیه ساجدی

پروژه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

تیر ۱۳۹۷

چکیده

با توجه به حجم بالای داده‌های غیر ساختاری تولیدشده توسط انسان که موجب عدم کارکرد مناسب موتورهای جستجو در ارائه‌ی پاسخ مناسب به سوالات است نیاز به یک سیستم طبقه‌بندی سوال مطرح می‌شود. در یک سیستم طبقه‌بندی سوال، هر سوال می‌تواند 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 دسته‌ای نسبت داد. هدف پروژه طراحی سیستمی برای طبقه بندی سوالات فارسی مبتنی بر روش وزن دهی عبارات است که در این پروژه ازروش‌های بهبودیافته‌ی TF [۱] و TF-IDF [۲] استفاده می‌شود که عبارتند از : mTF، mTF-IDF، mTF-mIDF، TF-mIDF . همچنین تاثیرگذاری این روش‌های پیشنهادی را با روش‌های دسته‌بندی مختلف همچون ماشین بردار پشتیبان [۳] و k نزدیک ترین همسایه [۴] مورد بررسی قرار میدهیم . ابتدا باید عملیات پیش پردازش انجام دهیم که شامل حذف کلمات کم‌معنی و بی‌معنی و علامت‌های نگارشی ، ریشه‌گیری ، بن‌واژه‌سازی و ... است . سپس با روش‌های پیشنهادشده به ویژگی‌ها وزن اختصاص دهیم. قابل ذکر است که انتخاب ویژگی یک فاز در دسته‌بندی متون است و به روش‌های وزن‌دهی وابسته است. ویژگی‌های انتخاب شده باید در یک قالب مناسب نمایش داده شوند . در این جا از مدل فضای برداری [۵] که یک مدل تاثیر گذار است بهره می‌بریم، در این مدل هر مستند به صورت یک بردار از ویژگی ها که متناظر با آن برداری از وزن آنها داریم نمایش داده می‌شود. در انتها به ارزیابی و ارائه نتایج حاصل میپردازیم.

[1] Term-frequancy
[2] Term Fequancy-Inverse Document Fequancy
[3] Support Vector Machine (SVM)
[4] K-Nearest Neighbor (KNN)
[5] 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 آنها که دسته بندی سوالات فارسی است ، اشاره می شود . طبقه بندی سوال به هیچ وجه کار بی اهمیتی نیست و نمی‌توان به سادگی و تنها با تکیه بر کلمات پرسشی به نتایج رضایت بخشی در زمینه ی دسته بندی سوالات رسید . ساختارهای متنوع نحوی و جمله‌بندی سوالات دسته‌بندی آنها 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 زمینه‌ی دسته بندی سوالات فارسی و انگلیسی انجام شده است. در بخش سوم به ارائه روش پیشنهادی برای وزن‌دهی به عبارات و همچنین به پیاده سازی سیستم طبقه بندی سوال بر پایه‌ی روش پیشنهادی می پردازد. در بخش چهارم به مقایسه ی روش پیشنهادی با سایر روش های وزن‌دهی به عبارات می‌پردازد . در نهایت در بخش آخر به جمع بندی و نتیجه‌گیری حاصل از این پروژه پرداخته می‌شود.

فهرست مطالب

فصل ۱

مفاهیم مقدماتی

۱.۱ معرفی ساختار 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 ارسال می‌شود تا با کلمات داخل خدمتگزار مقایسه شود. 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 ریشه یابی کرده ایم را در جدول ۱.۱ مشاهده می کنیم

[2]Server
[3]Stemming

سوال	سوال ریشه یابی شده
یونانیان به هگمتانه چه میگفتند	یونان به هگمتانه چه گفت
خط نابینایان چه نام دارد	خط نابینا چه نام داشت
اصطلاح فلوره در چه رشته ورزشی بکار میرود	اصطلاح فلوره در چه رشت ورزش کاشت رفت

جدول ۱.۱: نمونه هایی از ریشه یابی سوالات فارسی

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.

۲.۱ فراهم سازی داده مناسب

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 مهم از این عملیات می‌پردازیم:

۱.۲.۱ پالایش 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 آشنایی با برخی 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 [۹]
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 [۱۰]
در این روش الگوریتم یادگیری مدل و انتخاب ویژگی را به صورت توأمان انجام می‌دهد.[۶]

روش فیلتری [۱۱]
در این روش قبل از اجرای الگوریتم، انتخاب ویژگی انجام می‌شود و الگوریتم فقط یادگیری مدل را انجام می‌دهد.[۶]

روش انحصاری [۱۲]
در این روش فقط الگوریتم وظیفه انتخاب ویژگی را دارد و یادگیری مدل را انجام نمی‌دهد. یکی از کاربردهای مهم انتخاب زیرمجموعه ویژگی‌ها، تمرکز بر برخی ویژگی‌ها جهت یافتن ارتباط میان آنهاست.[۶]

۳.۲.۱ انتخاب ویژگی از سوال بر اساس اطلاعات زبان شناختی

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 [۱۳] ، نحوی [۱۴] و معنایی [۱۵] 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 عبارتند از:

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 معرفی برخی روش های طبقه 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. در این روش میزان کارایی با افزایش تعداد ویژگی ها، کاهش می‌یابد.
این ضعف به دلیل این است که این روش فاز واقعی یادگیری ندارد و باعث می‌شود هزینه‌ی محاسبات افزایش یابد.

[19] Lazy Classifier

۴.۱ کارهای پیشین در زمینه طبقه بندی سؤال

برای سیستم طبقه بندی سوال سه رویکرد پیشنهاد می شود :

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

روش اول سعی میکند تا سوال ها را با قوانین دستی که از قبل ساخته شده اند متناظر کند تا بتواند نوع پاسخ سوال را شناسایی کند اما نوشتن این قوانین به صورت دستی کاری ملال آور است و باعث می شود 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 [۲۰] (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 [۲۱] 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]

[20] Li and Roth
[21] Krishnan

هوانگ [۲۲] 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 [۲۳] را پیشنهاد میکنند و در مدل های ماشین بردار پشتیبان خطی و بیشترین آنتروپی [۲۴] به ترتیب به دقت های ۸۹.۲ و ۸۹.۰ درصد برای ۵۰ کلاس ، دست یافتند.[۳]

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.

[22] Huang
[23] Head Word
[24] Maximum Entropy (ME)
[25] Olalere Williams

فصل ۲

سیستم طبقه بندی سوال بر مبنای روش های 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

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

نظر می گردد. از تمام واژه‌های موجود در مستندات ، یک مجموعه واژگان به وجود می آید. 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 کیسه واژگان

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 [1] 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 .

[1] Bag of words

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. 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، در برگیرد. 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 می کنند. 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

[2] Training model

مداوم اعداد بردارهای لغات، نهایتا به یک وضعیت باثبات می رسد که همان بردارهای مورد بحث ماست.

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 اسکیپ گرام 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 های بهبود یافته TF-IDF برای وزن دهی به ویژگی ها

در این بخش به معرفی روش های بهبود یافته TF-IDF برای وزن‌دهی به عبارات می‌پردازیم. همان طور که اشاره شد این روش ها برای بهبود عملکرد دسته بندی متون معرفی شده‌اندو عبارتند از:[۱]

mTF-mIDF mTF-IDF,mTF,TF-mIDF این روش های پیشنهاد شده عبارت های گم شده را در محاسبه ی وزن عبارت های موجود به حساب می‌آورند.عبارت گم شده عبارتی است که به عنوان ویژگی در نظر گرفته شده است ولی در مستند حضور ندارد که دلیل آن میتواند کوتاه بودن مستند و یا ارتباط ضعیف بین ویژگی و دسته باشد. تعداد محدودی از متد ها هستند که برای وزندهی به عبارت های موجود،عبارت های گم شده را نیز در نظر میگیرند از این متد ها می توان به BTWS اشاره کرد. این متد برای عبارت های غایب در یک مستند وزن در نظر میگیرد و وزنی که به آن‌ها اختصاص میدهد مشابه عبارتی است که تنها یک بار در مستند ظاهر شده است. دسته بندی متن به معنی این است که هر مستند را به یک کلاس که از قبل تعریف و برچسب گذاری شده است نسبت بدهیم. نحوه ی نمایش متون گام مهمی است که دسته بند بتواند با محتوای متنی مستند سرو کار داشته باشد.مدل فضای برداری یک مدل پرکاربرد برای نمایش متن است که در آن هر مستند به صورت برداری از ویژگی های وزن دار در نظر گرفته می شود.ویژگی ها می‌توانند از نوع های مختلفی باشند برای مثال از کلمات، عبارت های نحوی و معنایی و همچنین بخشی از متن می‌توان به عنوان ویژگی استفاده کرد. اهمیت وزن هر ویژگی در مستندات یا مجموعه داده

نمایان می‌شود. عملکرد دسته بند تحت تاثیر وزن تخصیص داده شده به ویژگی ها می باشد این موضوع سبب شده است که توجه ما به متد های وزن دهی عبارات جلب شود. متد های وزن‌دهی به عبارت را میتوان به دو دسته ی متدهای با نظارت و بدون نظارت تقسیم کرد.تفاوت این دو دسته در استفاده یا عدم استفاده از اطلاعات مربوط به دسته بندی کلاس ها است. روش های با نظارت برای محاسبه‌ی وزن عبارت ها از اطلاعات کلاسی که مستند به آن تعلق دارد استفاده میکند. اطلاعات آماری که در روش های وزن دهی بدون نظارت مورد استفاده قرار میگیرد شامل تعداد مستندات،طول آنها،فرکانس هر عبارت در سطح مجموعه داده و غیره می‌باشد. شماهای پیشنهادی ویژگی های تاثیرگذار تر را برای دسته بندی متون انتخاب می‌کنند و محاسبات آماری نشان داده‌است که عملکرد دسته بند را بهبود می‌بخشند. متد mTF سهم تعداد عبارت‌های غایب در مستند را به کل عبارت‌های مجموعه داده در نظر می‌گیرد و همچنین متد mIDF سهم تعداد مستندات وقتی که یک عبارت در آن‌ها حضور ندارد به کل مستندات مجموعه داده در نظر میگیرد و شمای استاندارد وزن‌دهی TF-IDF بر پایه‌ی این شماهای mTF,mIDF ارائه مشوند.

Collection Frequency(CF) مجموع همه‌ی رخدادهای یک عبارت در همه‌ی مستندات می‌باشد.
Document Frequency(DF) تعداد مستنداتی است که آن عبارت در آن‌ها ظاهر شده‌است.
Document Length(DL) بیانگر تعداد عبارت‌های یک مستند بدون در نظر گرفتن تکرار آن‌هاست.

CF,DF,DL برای محاسبه‌ی وزن عبارات در شما های مختلف وزن‌دهی مورد استفاده قرارمی‌گیرند. در این بخش ۵ شمای وزن دهی به عبارات معرفی شده‌است که هیچ کدام از این متدها ، عبارت های گم شده را در وزن‌دهی به عبارات موجود در نظر نمی‌گیرند. همه‌ی این متد ها به جز term frequency relevance frequeny(tf-rf) بدون نظارت هستند ، این متد اخیرا پیشنهاد شده و عملکرد خوبی در طبقه بندی متون را نشان می‌دهد. TF یک شمای وزن‌دهی محلی است که وزن نرمال شده‌ی هر ترم مشخص در مستند را نشان می‌دهد و با تقسیم تکرار عبارت t در مستند d بر نرم اقلیدسی به دست می‌آید و این متد در مستندات کوتاه که تعداد وقوع عبارت متناسب است با میزان اهمیت آن کاربرد دارد.

فرمول به صورت زیر می‌باشد:

$$TF_{t,d} = \frac{f_{t,d}}{\sqrt{\sum_1^n f_{t,d}^2}} \tag{۱.۲}$$

DF یک شمای وزن‌دهی عمومی است و نشان دهنده‌ی تعداد مستنداتی است که یک عبارت مشخص در آن‌ها ظاهر شده‌است البته بدون در نظر گرفتن تکرار عبارات. این متد فرض می‌کند عبارت‌هایی که در تعداد بیشتری از مستندات ظاهر شده اند از اهمیت بیشتری برخور دارند. محتمل است عبارت های مختلف دارای DF یکسان باشند. فرمول به صورت زیر است:

$$DF_t = \sum_1^N \begin{cases} 1 & t \in d \\ 0 & o.w \end{cases} \tag{۲.۲}$$

DF,IDF در سطح مجموعه داده و متد های محلی در سطح مستندات وزن‌دهی انجام می‌دهند. متد TF-IDF بازتاب این فرض است که عبارت هایی که مهمتر هستند اون عبارت هایی هستند که DF کمتر دارند و بالعکس. برای کاهش حساسیت از تابع لگاریتمی استفاده می‌شود زیرا وقتی یک عبارت در یک مستند n برابر یک عبارت دیگر ظاهر شود به این معنی نیست که n برابر آن اهمیت دارد. فرمول به صورت روبه‌رو است:

$$TF - IDF_{t,d} = TF_{t,d}.IDF_t \tag{۳.۲}$$

where

$$IDF_t = log(\frac{N}{DF_t}) + 1 \tag{۴.۲}$$

N تعداد مستندات مجموعه داده است. متد Glasgow متدی است که از غلبه ی عبارات در مستندات طولانی جلوگیری می‌کند زیرا عبارت های کم اهمیت در چنین مستنداتی بسیار یافت می‌شود. این متد به عبارتی کم تکرار که در یک مستند کوتاه یافت شده است نسبت به عبارتی که در مستندی طولانی‌تر قرار دارد وزن بیشتری اختصاص می‌دهد. این شما شبیه به TF-IDF است زیرا هر دو از مقدار عمومی IDF برای وزن‌دهی به عبارات

استفاده می‌کنند با این تفاوت که نرمال‌سازی در TF-IDF با تقسیم بر طول مستند انجام شده است و در Glasgow با تقسیم بر نرم اقلیدسی. فرمول به صورت روبه‌رو است:

$$w_{td} = \frac{log(f_{td}+1)}{log(length_t)} \times (log(\frac{N}{DF_t})+1) \qquad (۵.۲)$$

tf.rf به عنوان یک شمای وزن‌دهی با نظارت معرفی شده است این متد از توزیع مستندات و تعداد نمونه های کلاس مثبت و منفی در محاسبه‌ی وزن عبارات استفاده میکند. فرمول به صورت روبه رو است:

$$tf.rf_{td} = tf_{td}.log(2+\frac{a}{max(1,c)}) \qquad (۶.۲)$$

a تعداد مستندات در کلاس مثبت که شامل عبارت t هستند و c تعداد مستنداتی که در کلاس منفی شامل ترم t هستند را نشان می‌دهد.

ایده‌ی اصلی شماهای پیشنهادی این است که فاکتوری به عنوان نماینده‌ی عبارت های گم شده و همچنین تعداد مستنداتی که عبارت مشخص در آن ها حضور ندارد را در محاسبه ی وزن عبارت های موجود شرکت دهیم. فرمول mTF به صورت زیر قابل مشاهده است:

$$mTF_{t,d} = \frac{tf_{t,d}.log(\frac{\sqrt{T_c}}{T_t})}{log[(\sum_{t=1}^{n} tf_{t,d}^2).(\frac{length_d^2}{\sqrt{T_c}})]} \qquad (۷.۲)$$

where

$$T_t = \sum_{d=1}^{D} tf_{t,d} \quad where \quad tf_{t,d} > 0 \qquad (۸.۲)$$

and

$$T_c = \sum_{d=1}^{D} \sum_{t} tf_{t,d} \tag{۹.۲}$$

T_t بیان‌گر تکرار یک عبارت مشخص در کل مستندات و T_c بیان‌گر تعداد عبارت‌های کل مجموعه داده است. $Length_d$ طول یک مستند است و تعداد عبارت‌های متمایز در مستند d است. در این متد پیشنهادی نسبتی بین تکرار عبارت در مجموعه داده و همه ی عبارت های متمایز کننده در مجموعه داده درنظر گرفته می‌شود تا با توجه به عبارت های غایب روی وزن عبارت های موجود تاثیرگذار باشد به علاوه نسب تعداد عبارت‌های متمایز در یک مستند به تعداد عبارت‌های متمایز کننده در مجموعه داده هم در نظر گرفته می‌شود.

در متد تغییریافته ی mIDF تعداد مستنداتی که عبارت t در آن ها ظاهر نشده‌است هم مورد استفاده قرار می‌گیرندکه اگر N تعداد کل مستندات باشد N-DFt تعداد مستنداتی است که عبارت t درآن ها ظاهر نشده است. فرمول به صورت زیر است:

$$mIDF_t = log[\frac{N}{\frac{1}{((N-DF_t)+1)}}] \tag{۱۰.۲}$$

بعد از ساده کردن داریم:

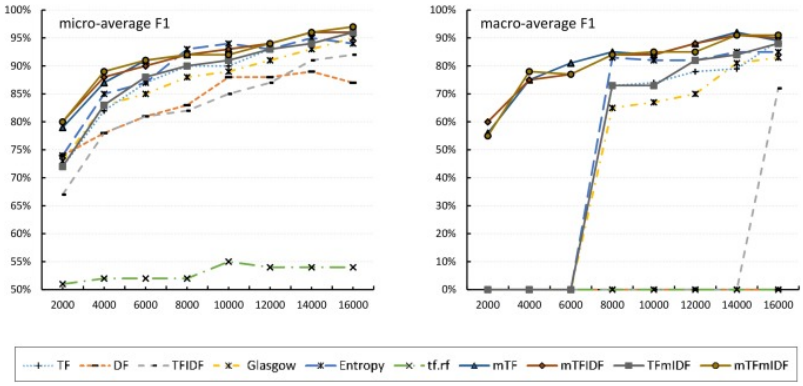
$$mIDF_t = log(N^2 - NDF_t + N) \tag{۱۱.۲}$$

سه فرم دیگر نیز قابل استخراج هستند.

$$mTFIDF_{t,d} = mTF_{t,d}.IDF_t \tag{۱۲.۲}$$

$$TFmIDF_{t,d} = TF_{t,d}.mIDF_t \tag{۱۳.۲}$$

$$mTFmIDF_{t,d} = mTF_{t,d}.mIDF_t \tag{۱۴.۲}$$



شکل ۱.۲: مقایسه‌ی کارایی شش شمای وزن‌دهی مختلف با دسته‌بند SVM روی مجموعه داده‌ی Reuters-21578

۴.۲ تفاوت کارایی روش های پیشنهادی با دیگر شماهای وزن‌دهی به عبارات

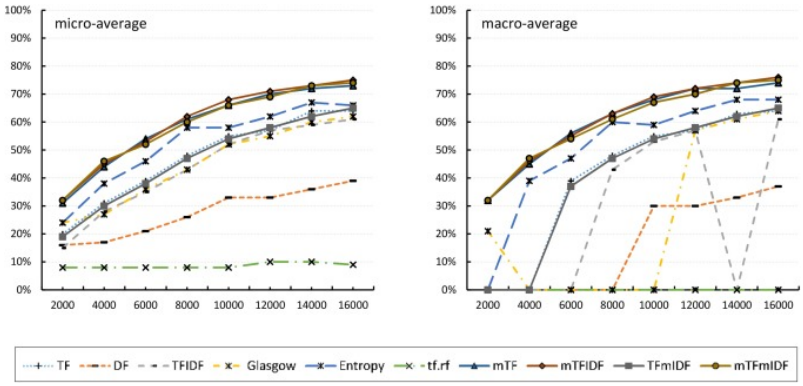
در این قسمت به تحلیل نتایج آزمایشات روی مجموعه داده‌های مختلف وقتی ازانواع متدها برای وزن‌دهی عبارات و از دسته بند ماشین بردار پشتیبان استفاده شده است، می‌پردازیم. معیار های ارزیابی را micro-average و macro-average در نظر می‌گیریم.[۱]

نتایج آزمایشات روی مجموعه داده Reuters-21578(R8)

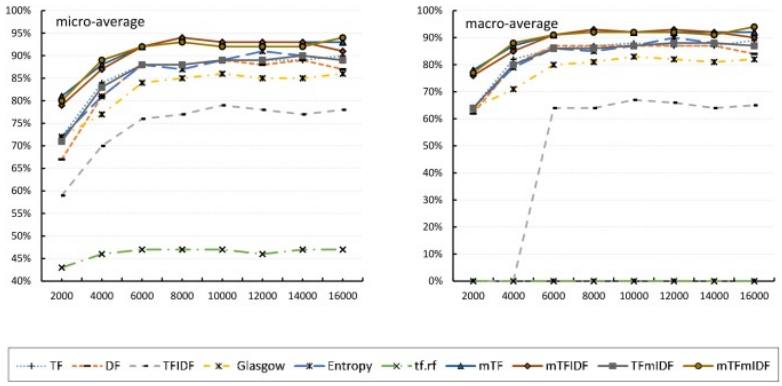
وقتی تعداد ویژگی‌ها افزایش یابد کارایی برای همه‌ی شما های وزن‌دهی افزایش می‌یابد. این مجموعه داده دارای ۱۰ دسته است که بعد از انجام عملیات پیش پردازش به ۷۶۷۴ مستند در داخل ۸ دسته تبدیل می‌شود.توزیع مستند ها درکلاس‌های این مجموعه داده ناهمواراست. شماهای پیشنهادی وقتی که تعداد ویژگی ها به ۱۶۰۰۰ می رسد بالاترین کارایی را دارند(حدود ۹۷ درصد) شمای آنتروپی هم با آن‌ها در رقابت است همچنین شماهای mTF,mTF-IDF,mTFmIDF برای تعداد ویژگی کمتر از ۶۰۰۰ هم عملکرد خوبی را نشان می دهند در حالیکه عملکرد شما های دیگر در این بازه بسیار ضعیف است.[۱]

نتایج آزمایشات روی مجموعه داده‌ی 20 Newsgroup

این مجموعه داده ۲۰۰۰۰ تا مستند دارد که در ۲۰ دسته قرار می‌گیرند بعد از عملیات فیلترینگ و حذف رکورد های تکراری تعداد مستندات به ۱۸۸۲۱ می‌رسد همچنین شاهد



شکل ۲.۲: مقایسه‌ی کارایی شش شمای وزن‌دهی مختلف با دسته‌بند SVM روی مجموعه داده‌ی 20 Newsgroup



شکل ۳.۲: مقایسه‌ی کارایی شش شمای وزن‌دهی مختلف با دسته‌بند SVM روی مجموعه داده‌ی WebKB

این موضوع هستیم که شمای آنتروپی دیگر در رقابت با شماهای پیشنهادی نمی‌باشد. عملکرد همه‌ی شماها در این مجموعه داده کمتر از عملکرد آن‌ها در مجموعه داده‌ی قبلی است که علت آن می‌تواند تعداد زیاد دسته های این مجموعه داده باشد.[۱]

نتایج آزمایشات روی مجموعه داده‌ی WebKB

این مجموعه داده ۴۱۹۹ مستند دارد که در ۴ دسته قرار می‌گیرند. نتایج عملکرد شما های وزن دهی روی این مجموعه داده بیشتر از عملکرد آن‌ها روی مجموعه داده‌ی 20 Newsgroup است به این دلیل که تعداد کلاس‌ها کمتر است.[۱]

فصل ۳

روش های تحلیل و آزمایش مدل

۱.۳ معیارهای تفسیر عملکرد دسته‌بند

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 : تعداد مستندات غیرمتعلق به یک کلاس که به غلط در آن کلاس شناسایی شده اند.
TP : تعداد مستندات متعلق به یک کلاس که به درستی در آن کلاس شناسایی شده اند.
FN : تعداد مستندات متعلق به یک کلاس که به غلط در آن کلاس شناسایی نشده اند.
TN : تعداد مستندات غیرمتعلق به یک کلاس که به درستی در آن کلاس شناسایی نشده اند.

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 مدل

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 فراهم سازی و انتخاب 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.

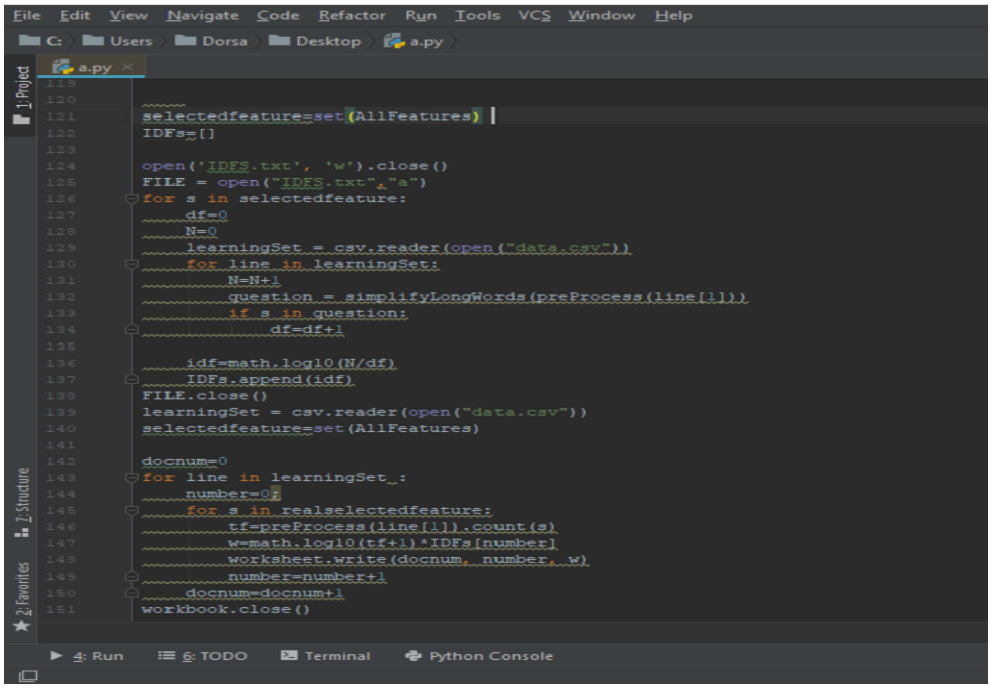
۲.۴ تبدیل به بردار

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 که ستون آخر هم برچسب آن سوال است. 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 استفاده از روش TF-IDFبرای وزن‌دهی

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 و با دسته‌بند SVM ، ۷۹.۳ درصد است.

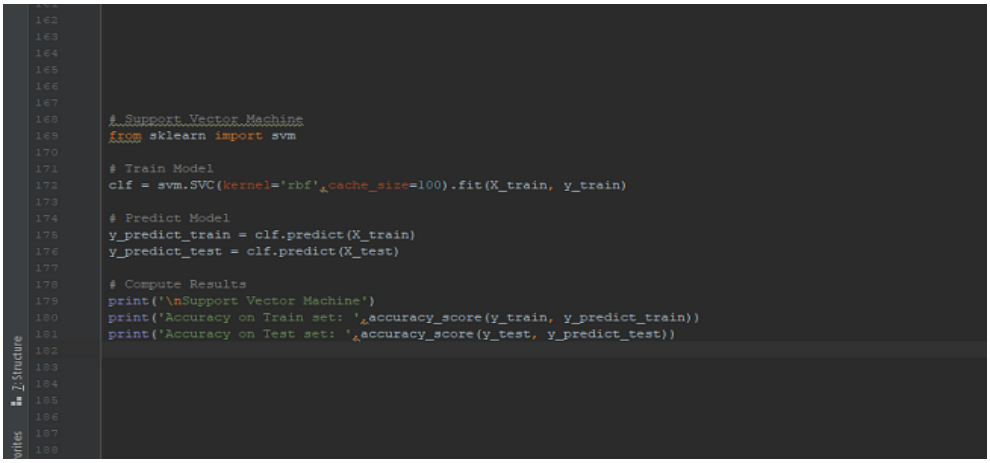
```
# KNeighborsClassifier
from sklearn.neighbors import KNeighborsClassifier

# Train Model
clf = KNeighborsClassifier(n_neighbors=15,algorithm='ball_tree').fit(X_train, y_train)

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

# Compute Results
print('\nKNeighborsClassifier')
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 ۲.۴: کد پایتون برای دسته‌بند kNN

۴.۴ استفاده از نرم افزار رپیدماینر برای طبقه‌بندی

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

۵.۴ مقایسه روش TF-IDFبا روش TFmIDF

در ادامه با روش TFmIDF وزن‌دهی به عبارات را انجام دادیم که کد آن در شکل ۶.۴ قابل مشاهده است بعد از دسته‌بندی به روش SVM دقت ۸۳.۸ درصد به دست آوردیم که پیشرفت خوبی نسبت به TF-IDF را دارد. هم‌چنین با دسته‌بند kNN دقت را به ۸۴.۶ درصد رساندیم. از این آزمایش‌ها می‌توان نتیجه گرفت روش‌های وزن‌دهی پیشنهاد شده تاثیر خوبی در افزایش دقت دسته‌بندی دارند.

	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

```
143
144         idf = math.log10(math.pow(N,2)-N*df+N)
145         IDFs.append(idf)
146     FILE.close()
147     learningSet = csv.reader(open("data.csv"))
148     selectedfeature = set(AllFeatures)
149
150     docnum = 0
151     for line in learningSet:
152         number = 0;
153         for s in realselectedfeature:
154             tf = preProcess(line[1]).count(s)
155             w = tf * IDFs[number]
156             worksheet.write(docnum, number, w)
157             number = number + 1
158         docnum = docnum + 1
159     workbook.close()
160
161
```

شکل ۶.۴: کد پایتون برای وزن‌دهی به روش TFmIDF

منابع

[1] Sabbah.T, Salamat.A, Selamat.Md, Herrera.E, S.Al-Anzi.F, Fujita.H, and Krejcar.O. " Modified frequency-based term weighting schemes for text classification." Applied Soft Computing,vol.58 (2017) (Pages 193-206)

[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 modified TF-IDF term weighting methods in Persian question classification.

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

Question classification in Persian using modified TF-IDF term weighting

Dorsa Kehaghi

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