اولین کنفرانس ملی نوآوری در مهندسی کامپیوتر و فناوری اطلاعات
1st National Innovation Conference on Computer Engineering and Information Technology
SHAFAGH UNIVERSITY 12/DECEMBER/2013
مؤسسه آموزش عالی غیردولتی غیرانتفاعی شفق تنکابن زمان: ۲۱ آذر ۱۳۹۲

دسته بندی متون به کمک الگوریتم KNN بهبودیافته مبتنی بر خوشه‌های وزن‌دار

زهرا صفیان بلداجی[۱]، محمدنادری دهکردی[۲]، حمیدرستگاری[۳]
[۱]دانشجوی کارشناسی ارشد مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد نجف آباد، نجف آباد
z_safian@yahoo.com
[۲] استادیار گروه کارشناسی ارشد نرم افزار،دانشگاه آزاد اسلامی واحد نجف آباد، نجف آباد
naderi@iaun.ac.ir
[۳]استادیار گروه کارشناسی ارشد نرم افزار ،دانشگاه آزاد اسلامی واحد نجف آباد، نجف آباد
rastegari@iaun.ac.ir

خلاصه

دسته بندی متون یکی از شاخه‌های متن‌کاوی است که به طور خودکار متون را طبقه‌بندی می‌نماید. تاکنون روشهای مفید بسیاری با رویکرد رده‌بندی مبتنی بر ناظر برای دسته‌بندی متون معرفی شده‌اند. در بین این روشها الگوریتم K نزدیکترین همسایه یا KNN به طور وسیع مورد استفاده قرار می‌گیرد،که به علت سادگی و بی‌پارامتر بودن و کارایی دسته‌بندی(دقت دسته‌بندی) آن یکی از بهترین روشها است. اما روش سنتیKNN پیچیدگی محاسباتی بالایی دارد و این نقیصه از کاربردی بودن KNN می‌کاهد. خوشه-بندی یک راه حل برای این مشکل است، به این روش که تعداد نمونه های یادگیری در دسته‌بندی را کاهش می دهد و به این طریق ازپیچیدگی زمانی محاسبات می‌کاهد. در این مقاله ، ما یکی از روشهای ساده مبتنی برای خوشه‌بندی الگوریتم KNN راکه قبلا پیشنهاد شده است، انتخاب کرده‌ایم .این روش بوسیله خوشه‌بندی پیچیدگی زمانی را تا حد زیادی کاهش می‌دهد ولی مقداری از کارایی دسته‌بندی می‌کاهد. ما یک مقدار وزنی به مرکز خوشه‌ها نسبت می‌دهیم و همچنین مقدار K در الگوریتم KNN را کاهش می‌دهیم. آزمایشها بر روی مجموعه داده محک Reuter-21578 نشان می‌دهد که این روش می‌تواند کارایی دسته‌بندی را افزایش دهد.

کلمات کلیدی

دسته‌بندی متون، K نزدیکترین همسایه، خوشه‌بندی، وزن‌دهی

۱- مقدمه

با رشد سریع اینترنت و افزایش سریع حجم اطلاعات متنی، دسته‌بندی خودکار متون به یک تکنولوژی کلیدی در پردازش و سازماندهی حجم وسیع مستندات تبدیل شده است. دسته‌بندی متون به فرآیند اختصاص یک سند متنی به یک یا چند دسته از پیش تعیین شده بر اساس محتوای آن سند و مجموعه متن‌های آموزشی که از قبل دسته-بندی شده‌اند، گفته می‌شود. امروزه از دسته‌بندی متون در بسیاری از زمینه‌ها مانند دسته‌بندی صفحات وب، فیلتر کردن هرزنامه در ایمیل ها و فهرست‌بندی متون و مانند آنها استفاده می‌شود.

تاکنون الگوریتم‌های مفید و مشهور متعددی برای دسته‌بندی متون معرفی شده است ازجمله ماشین‌های بردار پشتیبان، روش‌های بیز، K نزدیکترین همسایه و شبکه‌های عصبی ،درخت‌های تصمیم-گیری و مانند آنها.

در بین این روش‌ها K نزدیکترین همسایه یا به اختصار KNN به علت سادگی و کارایی آن یک روش پراستفاده دسته‌بندی می-باشد.KNN ساده‌ترین الگوریتم یادگیری ماشین می‌باشد: وقتی که یک نمونه جدید می‌آید KNN در بین نمونه های یادگیری K نزدیکترین همسایه به نمونه جدید را بوسیله معیارهای مناسب اندازه گیری شباهت، می‌یابد. سپس آن دسته‌ای که در بین K نزدیکترین همسایه تعداد متون بیشتری را به خود اختصاص داده است به عنوان دسته یا برچسب نمونه جدید تعیین شود.

دسته‌بندی های KNN یاد گیرنده‌هایی تنبل هستند، چرا که مدلها مانند یاد گیرنده‌های مشتاق (مانند درخت‌های تصمیم‌گیری، ماشین های بردار پشتیبان و مانند آن) به طور صریح ساخته نمی‌شوند. بنابراین ساختن مدل بسیار ارزان ولی دسته‌بندی یک شی جدید به نسبت گران است چرا که نیاز به محاسبه K نزدیکترین همسایه شی جدید که می‌خواهد دسته‌بندی شود داریم و باید فاصله (شباهت) این شی با همه اشیاء که از قبل دسته بندی شده اند محاسبه شود که این کار به ویژه برای مجموعه های یاد گیری که بزرگ هستند بسیار پرهزینه است، چه از لحاظ هزینه زمان و چه حافظه مورد نیاز آن. روشهای بسیاری برای ارتقای سرعت جستجو در KNN پیشنهاد شده است که عمدتا در دو دسته قرار می‌گیرند:

یک: طراحی الگوریتم‌های سریع برای جستجوی K نزدیکترین همسایه متن تست در کوتاهترین زمان.

دوم: انتخاب برخی نمونه های نماینده از نمونه‌های یادگیری اولیه یا حذف برخی ازنمونه‌های یادگیری اولیه.

برای دسته اول مرجع[5] ساختار درختی را پیشنهاد می‌کند، مرجع[6] نیز از تکنولوژی P-Tree استفاده می‌کند. پیچیدگی ساختن درخت مشکل اصلی این روش‌هاست.

برای دسته دوم مرجع[4] یک روش مبتنی بر چگالی که سعی می‌کند تعداد نمونه‌ها در نواحی چگال تر را کاهش دهد تا به چگالی متوسط در کل مجموعه یاد گیری برسد. این روش تا حدی حجم محاسبات را کاهش می دهد.

روش‌های مبتنی بر خوشه بندی در [9،8،7] به عنوان روش‌هایی دیگر در دسته دوم مطرح هستند.

مرجع [7] از خوشه‌بندی برای یکنواخت کردن چگالی مجموعه یادگیری استفاده می‌کند به این ترتیب که نقاطی که در یک شعاع همسایگی مشخصی قرار دارند و همگی متعلق به یک دسته هستند در یک خوشه قرار می‌گیرند و مرکز خوشه به عنوان نماینده آنها نشان داده می‌شود. در این روش زمان اجرا و دقت تابعی از شعاع انتخابی است.

مرجع [8] از الگوریتم خوشه بندی K-means استفاده میکند و نقاط اولیه مراکز خوشه را بر اساس چگالی نواحی مختلف مجموعه یادگیری انتخاب می‌کند و به این طریق از ایجاد بهینه محلی یا انتخاب نقاط منفرد به عنوان مراکز اولیه خوشه ها جلوگیری می‌کند اما عیب آن روش پیچیده محاسبات چگالی نقاط یادگیری است.

مرجع[9] از یک روش ساده برای خوشه بندی مجموعه نمونه های یاد گیری استفاده می‌کند و بوسیله استفاده از مراکز خوشه ها بعنوان مجموعه یاد گیری یک روش بسیار سریع برای الگوریتم KNN ارایه می‌دهد.اما مقداری جزیی از دقت دسته‌بندی را کاهش می‌دهد. در این مقاله ما یک مقدار وزنی برای مراکز خوشه ها معرفی می‌کنیم که می‌تواند علاوه بر پیچیدگی زمانی کم (کارایی زمانی) بتواند دقت (کارایی دسته‌بندی) را نیز افزایش دهد.

ادامه مقاله به صورت زیر سازماندهی شده است: در بخش دوم الگوریتم سنتی KNN برای دسته‌بندی متون شرح داده شده است. در بخش سوم روش الگوریتم KNN مبتنی بر خوشه‌بندی معرفی شده است و روش وزن‌دهی پیشنهادی برای الگوریتم KNN مبتنی بر خوشه‌بندی در بخش چهارم شرح داده شده است و نتایج آزمایشات در بخش پنجم و نتیجه گیری در بخش ششم مقاله آورده شده است.

۲- الگوریتم KNN برای دسته بندی متون

برای دسته‌بندی یک متن (سند) آن را معمولا به فرمت یک بردار نشان می‌دهیم. اجزای بردار را مجموعه‌ای از کلمات مهم که ویژگی نامیده

می‌شوند، تشکیل می‌دهند. این مدل، مدل فضای برداری نامیده می‌شود و KNN یک روش دسته بندی مبتنی بر این روش می‌باشد[3].

K نزدیکترین همسایه یا به اختصار KNN یک روش ساده یا به عبارتی ساده‌ترین الگوریتم یادگیری ماشین می‌باشد و از لحاظ تئوری یک روش بالغ می‌باشد. به زبان رسمی الگوریتم KNN به شرح زیر انجام می‌شود:

۱- همه اسناد یادگیری $(d_1,d_2,\ldots,d_n)$ در حافظه ذخیره می‌شوند.

۲- وقتی یک سند x وارد می‌شود، اسناد یادگیری برطبق اندازه شباهتی که با سند x دارند رتبه بندی می‌شوند. سپس K سندی که بیشترین شباهت را دارند ، انتخاب می‌شوند.

اندازه شباهت معمولا با روش فاصله کسینوسی بر طبق زیر تعیین می‌شود:

$$sim(x,d_j) = \frac{\sum_{k=1}^{m} x_k \times d_{jk}}{\sqrt{\sum_{k=1}^{m} x_k^2 \sum_{k=1}^{m} d_{jk}^2}} \qquad (1)$$

m تعدادکل ویژگی هاست.

۳- با استفاده از K سندی که بیشترین شباهت را به سند x دارند ، درجه تعلق سند x به هر دسته C_j بر طبق فرمول زیر محاسبه می‌شود:

$$p(x, C_j) = \sum_{d_i} sim(x, d_i) y(d_i, C_j) \qquad (2)$$

که $d_i \in KNN(x)$ یعنی K سند شبیه‌تر x را نشان می‌دهد و $y(d_i, c_j)$ تعلق سند x به دسته C_j را برطبق فرمول زیر مشخص می‌کند:

$$y(d_i, C_j) = \begin{cases} 1, d_i \in C_j \\ 0, d_i \notin C_j \end{cases} \qquad (3)$$

۴- در نهایت درجه تعلق سند x به هرکدام از دسته‌ها با هم مقایسه می‌شوند و x به دسته‌ای که بیشترین وزن $p(x, C_j)$ را دارد نسبت داده می‌شود.

$$C = \arg \max_j \left(p(x, C_j)\right) \qquad (4)$$

فرض کنید که N تعداد اسناد مجموعه یادگیری باشد و T تعداد اسناد آزمایشی و m تعداد ویژگی های انتخاب شده برای متون در نظر گرفته شده باشد، آنگاه پیچیدگی زمانی KNN برابر با O(TNm) می‌باشد که برای مجموعه های یاد گیری بزرگ، زمان اجرای بالایی را می‌طلبد.

مسائل مهمی وجود دارند که می‌توانند کارایی KNN را تحت تاثیر قرار دهند. یکی از آنها انتخاب K است.اگر K خیلی کوچک باشد نتیجه کار می‌تواند تحت تاثیر نقاط نویزی قرار گیرد. از طرف دیگر اگر K خیلی بزرگ باشد، آنگاه اسناد زیادی از سایر دسته ها در همسایگی سند جدید قرار می‌گیرند[10].

۳- الگوریتم KNN مبتنی برخوشه بندی

این روش که در مرجع[9] پیشنهاد شده است، از رویکرد خوشه‌بندی برای کاهش تعداد اسناد در مجموعه یادگیری استفاده می‌کند. این روش می‌تواند مقدار زیادی از حجم پیچیدگی محاسباتی الگوریتم KNN را کاهش دهد. این روش از هر دو روش خوشه‌بندی K-means و خوشه‌بندی سلسه مراتبی استفاده می‌کند تا مجموعه یادگیری را به بهترین روش خوشه‌بندی کند بنابراین در ابتدا یک معرفی خلاصه از این الگوریتم‌ها را ارایه می‌دهیم:

۳-۱- الگوریتم K-means

الگوریتم خوشه‌بندی K-means یک روش رایج خوشه‌بندی افرازی است و بسیار ساده و سریع می‌باشد. مراحل اصلی این الگوریتم عبارتند از:

گام۱: به صورت تصادفی K متن به عنوان نقاط مرکزی اولیه خوشه‌ها انتخاب می‌شود، K تعداد خوشه‌هاست.

گام۲: بر اساس میزان شباهت هر متن با هر مرکز خوشه، آن متن به خوشه شبیه‌ترین مرکز نسبت داده می‌شود و دوباره مراکز خوشه‌ها محاسبه می‌شود.

گام۳: گام۲ تا جایی‌که شرط تابع همگرایی برآورده شود، تکرار می‌شود.

۳-۲- خوشه بندی سلسله مراتبی

ایده خوشه‌بندی سلسله مراتبی بر این اساس است که در ابتدا هر شیئ داده شده به عنوان یک خوشه در نظر گرفته می‌شود که مرکز آن خوشه خود آن شیئ است. دو خوشه‌ای که بیشترین شباهت را به هم دارند با هم ترکیب می‌شوند تا زمانی که تعداد خوشه‌ها به K برسد. آنگاه مراکز هر خوشه محاسبه می‌شود. در طول این فرآیند اشیای متشابه به مرور در یک خوشه قرار می‌گیرند.

وقتی در خوشه Clu_1 و Clu_2 به یک خوشه ترکیب می‌شوند، بردار مرکزی این خوشه بوسیله بردار مرکزی Clu_1 و Clu_2 به دست می‌آید. فرض کنید تعداد اشیاء داده در Clu_1 برابر n_1 ودر Clu_2 برابر با n_2 باشد و بردار های مرکزی به ترتیب cen_1 و cen_2 باشند، بردار مشخصه خوشه جدید، cen به روش زیر به دست می‌آید:

$$cen = (n_1 \times cen_1 + n_2 \times cen_2)/(n_1 + n_2)$$

۳-۳- الگوریتم KNN مبتنی بر خوشه بندی

الگوریتم خوشه‌بندی K-means بسیار به نقاط اولیه مراکز خوشه‌ها حساس می‌باشد و انتخاب تصادفی آنها می‌تواند نتیجه را تاحد زیادی تغییر دهد چرا که ممکن است نقاط منفرد انتخاب شوند و یا حداقل دو نقطه در یک خوشه انتخاب شوند یا ممکن است مسئله بهینه محلی ایجاد شود. برای اجتناب از این مشکلات در ابتدا P×K متن به عنوان نقاط اولیه خوشه بندی در هر دسته انتخاب می‌شوند. الگوریتم K-means بر روی این مراکز اعمال می‌شود. سپس آن خوشه‌هایی که تعداد اعضای آن بسیار کم است مثلا نقاط منفرد که تعداد اعضای خوشه آنها ۱ عدد است، بیرون رانده می‌شوند تا از تاثیر مخرب آنها در خوشه‌بندی جلوگیری شود. سپس برای سایر مراکز خوشه‌ها، از روش سلسله مراتبی خوشه‌بندی برای ادغام دو خوشه‌ای که بیشترین شباهت را به هم دارند استفاده می‌شود و این کار تا زمانی که تعداد خوشه‌ها به عدد K برسد تکرار می‌شود و سپس K شی از مجموعه داده آموزشی که به مراکز خوشه‌ها نزدیکتر هستند به عنوان نقاط اولیه خوشه‌بندی انتخاب می‌شوند. سپس با استفاده از این K شی(متن) الگوریتم خوشه‌بندی K-means به طور جداگانه برای هر دسته از نمونه های آموزشی انجام می‌شود و بردار مرکزی هر خوشه محاسبه می‌شود. در نهایت K×|C| خوشه از کل مجموعه آموزشی ایجاد می‌شود، که |C| تعداد دسته‌ها می‌باشد.

وقتی یک متن نیاز به دسته بندی پیدا می‌کند الگوریتم KNN به جای استفاده از کل مجموعه آموزشی از این K×|C| مرکز خوشه برای محاسبه شباهت مجموعه آموزشی با متن جدید استفاده می‌شود و بوسیله این روش به مقدار قابل توجهی از زمان دسته‌بندی کاسته می‌شود.

۴- الگوریتم KNN مبتنی بر خوشه های وزن دار

الگوریتم KNN مبتنی بر خوشه بندی که در قسمت قبل به آن اشاره شد در کنار سادگی، عملکرد بسیار خوبی در کاهش پیچیدگی محاسباتی و همچنین مشکلات حافظه داشته است اما به مقدار کمی از دقت یا کارایی دسته‌بندی به روش KNN سنتی می‌کاهد. در این مقاله یک فاکتور وزن دهی برای مراکز خوشه‌ها پیشنهاد می‌کنیم که می‌تواند تا حدی دقت الگوریتم را افزایش دهد.

ایده اصلی این است که بوسیله خوشه‌بندی، همه خوشه ها وزن یکسانی در محاسبه شباهت نمونه آزمایشی (متن جدید) دارند اما با این روش ما وضعیت اولیه توزیع نمونه‌ها در مجموعه یاد گیری اولیه را از دست می‌دهیم و از تفاوت چگالی در خوشه های مختلف صرف نظر می‌کنیم که این ممکن است نتیجه دسته‌بندی را تغییر دهد بنابراین ما به هر مرکز خوشه یک مقدار وزنی متناسب با تعداد نمونه-های آن خوشه، نسبت می‌دهیم. اما از طرف دیگر این کار ممکن است به سوگیری دسته‌بندی به سمت خوشه‌هایی که چگالی آنها بالاتر است منجر شود و از اهمیت خوشه هایی که تعداد اعضای آنها کمتر است، بکاهد و حتی بدتر از آن احتمال دسته‌بندی برای دسته‌هایی که چگالی آنها کم است را کاهش دهد، چرا که مجموعه داده‌های نا متوازن در اسناد وب پدیده‌ای بسیار رایج است. بنابراین برای توازن این مقدار وزنی آن را در معکوس تعداد نمونه ها در دسته ای که مرکز خوشه به آن تعلق دارد، ضرب می‌کنیم.

مساله دیگری که باید به آن توجه شود این است که پس از خوشه‌بندی نمونه‌ها، ما به مقدار کمتری برای K در الگوریتم KNN نیاز داریم، چرا که نمونه‌های نزدیک به متن آزمایشی اکنون خوشه-بندی شده‌اند و تعدادی از آنها تنها با یک مرکز خوشه نشان داده شده‌اند، بنابراین می‌توانیم مقدار K در الگوریتم KNN را کاهش دهیم تا به این طریق کارایی دسته‌بندی را افزایش دهیم و هم چنین از پیچیدگی زمان محاسبات بکاهیم.

به زبان دقیق‌تر وزن‌دهی به روش زیر انجام می‌شود:

به عنوان نمونه دسته C_j را در نظر می‌گیریم، اگر بوسیله الگوریتم K-means به L خوشه تقسیم شود ما L خوشه $\{cl_{j1}, cl_{j2}, \ldots, cl_{jL}\}$ را به دست می‌آوریم. که برای نمایش دسته C_j به کار می‌روند. همچنین فرض می‌کنیم که تعداد نمونه‌ها دراین L خوشه به ترتیب برابر با $\{Num_{j1}, Num_{j2}, \ldots, Num_{jL}\}$ باشند، آنگاه مقدار وزنی W_{ji} برای هر مرکز cl_{ji} به صورت زیر محاسبه می‌شود:

$$W_{ji} = \frac{Num_{ji}}{N_{cj}} \qquad (5)$$

N_{cj} تعداد نمونه ها در دسته C_j را نشان می‌دهد.

اکنون یک تابع شباهت جدید برای هر مرکز خوشه و متن تست x به شرح زیر به دست می‌آوریم، که به جای فرمول (۱) استفاده می-شود.

$$sim(x, cl_{ji}) = \frac{w_{ji} \sum_{k=1}^{m} x_k \times cl_{jik}}{\sqrt{\sum_{k=1}^{m} x_k^2} \sqrt{\sum_{k=1}^{m} cl_{jik}^2}} \qquad (6)$$

سرانجام الگوریتم KNN مبتنی بر خوشه‌های وزندار به شرح زیر انجام می‌شود:

ورودی: مجموعه آموزشی TR و مجموعه تستی TE ، ابعاد ویژگی m و تعداد نزدیکترین همسایه K، تعداد مراکزخوشه در هردسته 'K.

خروجی: معیار F و نتایج دسته بندی

گام ۱: جداسازی کلمات، حذف کلمات کم‌ارزش ، ریشه‌یابی کلمات باقیمانده انجام شود و ماتریس کلمه-متن برای TRو TEساخته شود.

گام ۲: کاهش ابعاد داده ها برای TRوTE انجام شود تا m ، تعداد ویژگی ها به دست آید.

گام ۳: وزن هر ویژگی برای هر متن در TRوTE به روش *TF –IDF* محاسبه شود و همه متون به فرمت برداری نمایش داده شود.

گام ۴: برای هر دسته C_j در TR دو گام زیر انجام شود :

گام ۱: الگوریتم K-means برای به دست آوردن $P \times K'$ خوشه انجام شود.

گام ۲: خوشه‌بندی سلسله مراتبی به روشی که در بالا گفته شد، برای ادغام این $P \times K'$ خوشه به K’خوشه انجام شود.

گام ۵: برای هر متن x درTE شباهت بین بردار x و بردارمرکز خوشه Clu(Clu یک خوشه است) با فرمول تابع شباهت (۶) محاسبه شود. سپس مراکز خوشه به ترتیب شباهتشان مرتب شوند و K مرکزخوشه نزدیکتر مشخص شوند.

گام ۶: با استفاده از این K مرکز خوشه نزدیکتر به x، وزن دسته مربوط به x برای هر دسته C_j با فرمول های (۲)و(۳) محاسبه شود ودسته‌ای که x به آن تعلق دارد با فرمول (۴) مشخص شود.

گام ۸: معیار F و سایر نتایج دسته‌بندی مشخص شود.

۵- نتایج آزمایشات

۵-۱- مجموعه داده

مجموعه متون آزمایشی دسته‌بندی Reuters-21578، شامل اسنادی است که از روزنامه رویترز در سال ۱۹۸۷ جمع آوری شده است. این مجموعه یک محک استاندارد برای دسته‌بندی متون می‌باشد. در این مقاله ما از زیرمجموعه ModApte برای جداسازی مجموعه داده Reuters به مجموعه یادگیری و آزمایشی استفاده کردیم. بر اساس مستندات Reuters-21578، این زیر مجموعه باید ۹۶۰۳ سند در مجموعه یادگیری و ۳۲۹۹ در مجموعه آزمایشی را در بر گیرد. اما در بسیاری از این اسناد، موضوع(دسته) و متن بدنه وجود ندارد یا چند موضوع برای یک متن وجود دارد، بنابراین این دسته از اسناد را حذف کردیم. سرانجام ۱۱ دسته که تعداد متون آنها در بین کل دسته‌های مجموعه بیشتر است را برای مجموعه آموزشی استفاده کردیم. جدول۱ مشخصات مجموعه آموزشی و آزمایشی مورد استفاده در آزمایشات ما را نشان می‌دهد.

جدول(۱):مشخصات مجموعه آموزشی وآزمایشی مورد استفاده در آزمایشات

نام دسته	تعداد متن‌های آموزشی	تعداد متن‌های آزمایشی
gold	70	20
money-supply	70	17
coffee	89	21
sugar	90	24
ship	107	35
interest	140	57
money-fx	176	69
crude	223	98
trade	225	73
acq	1435	620
earn	2673	1040

۵-۲- معیار کارایی

برای ارزیابی کارایی دسته بندی از معیار رایج F1 استفاده کردیم. معیار F1 هردو معیار recall (یادآوری) و precision(دقت) را برای ارزیابی سیستم های دسته‌بندی متون مورد استفاده قرار می‌دهد. این معیارها به شرح زیر تعریف می‌شوند:

$$Recall=\frac{number\ of\ correct\ positive\ predictions}{number\ of\ positive\ examples}$$

$$Precision=\frac{number\ of\ correct\ positive\ predictions}{number\ of\ positive\ predictions}$$

$$F1=\frac{2\times Precision\times Recall}{Precision+Recall}$$

برای ارزیابی میانگین کارایی در کل دسته ها دو فرم میانگین معیار F1 وجود دارد[3]:

Macro- F1 = average of within-category F1 values.
Micro- F1 = F1 over categories and documents.

در آزمایشات ما معیار *F1*-Macro به عنوان معیار کارایی دسته-بندی در نظر گرفته شده است.

۵-۳- طراحی آزمایشات وتجلیل نتایج

برای انجام آزمایشات، هرکدام از اسناد از فرمت اولیه خود یعنی قالب sgml به بردار کلمات تبدیل شدند. در ابتدا تگ‌های sgml باید حذف شوند. این تگ ها را به وسیله Microsoft C# جدا کردیم وسپس هر سند را از فایل های Reuters استخراج کرده و هرکدام را در یک فایل جداگانه نوشتیم تا شاخص‌گذاری آنها ساده‌تر شود. سپس کلمات استخراج شدند و کلمات کم‌ارزش با استفاده از یک لیست کلمات انگلیسی پرتکرار حذف شده و در نهایت ریشه یابی کلمات به وسیله Porter stemmer انجام شد.

برای شاخص گذاری اسناد از وزندهی TF-IDF استفاده شد. انتخاب ویژگی با استفاده از روش آستانه تکرار اسناد (Document Frequency Thresholding) انجام شد وکلماتی که در کمتر از ۵ سند تکرار شده بودند با این فرض که کلمات نادر در تعیین دسته تاثیری ندارند، حذف شدند. بعد از جداسازی کلمات و حذف کلمات کم ارزش و انتخاب ویژگی، ابعاد ویژگی به ۳۵۳۰ رسید.

تعداد نمونه‌های اولیه برای دسته‌های gold ، money-supply ، coffee، sugar، ship، interest وmoney-fx نصف تعداد نمونه‌های دسته یادگیری مربوطه و برای دسته‌های crude و trade یک سوم تعداد نمونه‌های دسته و برای دسته acq یک چهارم تعداد نمونه‌های دسته و برای دسته earn یک پنجم تعداد نمونه‌های دسته انتخاب شد.

در الگوریتم‌های خوشه‌بندی K-means و سلسله مراتبی از اندازه شباهت کسینوسی برای اندازه‌گیری شباهت (فاصله) اسناد با مراکز خوشه‌ها استفاده کردیم. در این مقاله، ما برای دسته‌بندی KNNسنتی و KNN مبتنی بر خوشه بندی عدد K (تعداد همسایه-های نزدیکتر) را براساس نتایج [11] برابر با ۳۰ انتخاب کردیم و برای دسته‌بندی KNN مبتنی بر خوشه‌بندی وزندار، عدد K را برابر با ۱۵ قرار دادیم.

برای ارزیابی کارایی روش KNN مبتنی برخوشه‌بندی وزندار، آزمایشات برای تعداد خوشه‌های مختلفی استفاده شد تا نتایج ملموس-تر شود. جدول ۲ نتایج آزمایشات را نشان می‌دهد.

جدول(۲): نتایج آزمایشات

مقدار Macro-F1			
KNNمبتنی برخوشه-های وزندار	KNNمبتی برخوشه‌بندی	تعداد خوشه‌ها درهر دسته	KNNسنتی
0.81936	0.81902	k'=10	0.87432
0.80404	0.80255	k'=15	
0.81939	0.81085	k'=20	
0.83335	0.77882	k'=25	
0.83186	0.78879	k'=30	

با مشاهده جدول ۲ مشخص می‌شود که وقتی تعداد خوشه‌ها K'=10 روش KNN مبتنی برخوشه های وزندار کمی بهتر از KNN مبتنی برخوشه‌بندی عمل می‌کند. با افزایش 'K ، نتایج بهتر می‌شود تاجایی که به K'=25 میرسد و نسبتا کارایی خوبی تقریبا نزدیک به دسته بندی KNN به دست می‌آید. به هر حال پیچیدگی زمانی برای تعداد خوشه‌های بیشتر ، بالاتر است ، اما هنوز خیلی کمتر از KNN سنتی می‌باشد. به علاوه مقدار کمتر K برای تعداد همسایه های نزدیکتر اثر آن را تعدیل می‌کند. اما در هر حال این یک مصالحه بین کارایی دسته‌بندی و کارایی زمانی می‌باشد.

٦- نتیجه‌گیری

با رشد سریع اینترنت، دسته‌بندی متون به تکنولوژی کلیدی برای سازماندهی و پردازش حجم بالای داده‌های متنی تبدیل شده است. روش KNN به عنوان روشی ساده و موثر به طور وسیع در دسته بندی متون مورد استفاده قرار می‌گیرد. اما روش KNN نیاز محاسباتی بالایی را می‌طلبد. خوشه بندی روشی است که برای کاهش حجم داده‌های آموزشی مورد استفاده برای محاسبه شباهت با داده آزمایشی پیشنهاد شده است. در این مقاله یک مقدار وزنی براساس تعداد اسناد خوشه و تعداد اسناد دسته مربوط به مراکز خوشه‌ها برای بهتر کردن کارایی دسته‌بندی یک روش KNN مبتنی بر خوشه بندی که قبلا معرفی شده است، پیشنهاد شده و مقدار Kدر الگوریتم KNN برای پیش بینی دسته نمونه جدید کاهش داده شده است . آزمایشات نشان می‌دهد روش جدید کارایی (دقت) بهتری نسبت به روش KNN مبتنی بر خوشه بندی قبلی دارد. در آینده می‌توان این روش را بر روی سایر مجموعه دادها آزمایش نمود و همچنین سایر تکنیک‌های وزندهی به مراکز خوشه ها را بررسی کرده و کارایی را بیشتر افزایش داد.

مراجع

[1] [1] Chen T, Xie Y Q, *Review of Feature Reduction method in Text Categorization, [J]*,Journal of Information,24 (6): 690-684., 2005

[2] [2] Su Jinshu, Zhang Bofeng, Xu Xin, *Advances in Machine Learning Based Text Categorization, Journal of Software*, Vol.17, No.9, pp.1848□1859, 2006

[3] [3] Fang Lu, Qingyuan Bai, *A Refined Weighted K-Nearest Neighbors Algorithm for Text Categorization*, 978-1-4244-6793, IEEE,2010

[4] [4] Fang Yuan, Liu Yang, Ge Yu, *A New Density-Based Method For Reducing The Amount of Training Data In K-NN Text classification*, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 19-22 ,August 2007

[5] [5] Li Juan, *TKNN: an improved KNN algorithm based on tree structure*, IEEE, Seventh International Conference on Computational Intelligence and Security,2011

[6] [6] Imad Rahal , William Perrizo, *An Optimized Approach for KNN Text Categorization using P-trees*, ACM Symposium on Applied Computing,2004

[7] [7] Lijuan Zhou, Linshuang Wang, Xuebin Ge and Qian Shi, *A Clustering-Based KNN Improved Algorithm CLKNN for Text Classification*, IEEE 2nd International Asia Conference on Informatics in Control, Automation and Robotics,2010

[8] [8] Chun-Y an WANG, Yu-Guang YAN, Kuo Zhang and Jian-Gang Li, *A K-Nearest Neighbor Algorithm Based on Cluster in Text Classification*, IEEE, International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE),2010

[9] [9] Jialun Lin , Xiaoling Li and Yuan Jiao, *Text Categorization Research Based on Cluster Idea*, IEEE, Second International Workshop on Education Technology and Computer Science, 2010

[10] [10] XindongWu · Vipin Kumar And others, *Top 10 algorithms in data mining*, Springer-Verlag London Limited, 2007

[11] [11] Luigi Galavotti, Fabrizio Sebastiani, and Maria Simi, *Experiments on the use of feature selection and negative evidence in automated text categorization*, Proceedings of ECDL-00, 4th European Conference on Research and Advanced Technology for Digital Libraries, pages 59–68, Lisbon, PT, 2000. Springer Verlag, Heidelberg, DE. Published in the “Lecture Notes in Computer Science” series, number 1923.