به نام خدا

گزارش تمرین دوم درس NLP

دانشجو: محمد قربانی-۹۷۱۳۱۰۹۹

دانشگاه صنعتی امیرکبیر – دانشکده مهندسی کامپیوتر

بهار ۹۷-۹۸

روش اول (word2vec و استفاده از میانگین ساده):

در این روش ابتدا یک پیش پردازش اولیه بر روی داده ترین انجام صورت گرفت و کلمات ایست که تنها سربار محاسباتی به سیستم اضافه می‌کند حذف شدند. بعد از حذف کلمات ایست، مطالب را بر روی یک فایل ریخته و سپس بر روی آنها یک word2vec ساختیم. در ساخت مدل مقدار min_count را برابر با ۱ در نظر گرفتم که مشکل کلمات دیده نشده بوجود نیاید. سپس با استفاده از این مدل بردار کلمات هر داکیومنت را به‌دست آوردیم و سپس این بردارها را برای هر داکیومنت جمع کردیم و تقسیم بر تعداد کلمات آن داکیومنت‌ها کردیم و این بردار را به عنوان بازنمایش آن داکیومنت در نظر گرفتیم. بعد از ایجاد بردار بازنمایی هر داکیومن این بردارها را به الگوریتم کلاسترینگ خود دادیم تا مدل کلاسترینگ خود را ایجاد کنیم. بعد از ایجاد خوشه‌بند، بردار هر داکیومنت داده ترین خود را به خوشه‌بند دادیم تا آن را خوشه‌بندی کند. سپس بیشترین تعداد موضوع در هر خوشه را به عنوان لیبل آن خوشه در نظر گرفتیم و سپس به محاسبه NMI، Accuracy، F-Measure و V-Measure پرداختیم.

معیار	داده ترین	داده تست
NMI	0.56	0.58
Accuracy	0.72	0.72
F-Measure	0.73	0.74
V-Measure	0.56	0.58

NMI یکی دیگر از معیارهایی است که برای سنجش کیفیت خوشه‌بندی استفاده می‌شود. این معیار با در نظر گرفتن خوشه‌بندی واقعی آیتم‌ها و همچنین خوشه‌بندی تولید شده توسط الگوریتم شباهت بین این دو مجموعه از خوشه‌ها را بررسی می‌کند. این معیار بررسی می‌کند که این دو دسته از خوشه‌ها تا چه اندازه به هم مشابه و تا چه اندازه نسبت به هم متفاوت هستند. این معیار مقداری بین ۰ و ۱ دارد که مقدار ۱ بیانگر خوشه‌بندی ایده آل می‌باشد. مزیت معیار NMI این است که نسبت به تعداد خوشه-های در نظر گرفته شده نرمال می‌شود.

نتایج حاصل از این روش نشان می‌دهند که این روش دقت نسبتا خوبی داشته است.

روش دوم (word2vec و استفاده از میانگین وزن‌دار):

در این روش بردار هر کلمه در وزن tfidf آن کلمه ضرب شده است. مقدار tfidf به صورت زیر محاسبه می‌شود:

$$tf_{idf} = tf * idf$$

$$tf = \begin{cases} 1 + \log count(t,d) & if\ count(t,d) > 0 \\ 0 & else \end{cases}$$

$$idf = \log(\frac{N}{df_i})$$

که در آن N بیانگر تعداد کل اسناد موجود در پیکره است و df_i بیانگر تعداد داکیومنت‌هایی که کلمه مورد نظر را دارند است. استفاده از Idf موجب می‌شود که مشکل کلمات ایست که سربار محاسباتی دارند و هیچ گونه اطلاعات مفیدی نمی‌دهند حل شود. کلمات ایست حاصل idf کمی می‌گیرند و این باعث می‌شود که در محاسبات ارزش کمی بگیرند.

نتایج حاصل از این روش به صورت زیر می‌باشد. در مقایسه با روش اول کمی کاهش اندازه معیارها را نشان می‌دهد. دلیل این امر این است که ما در روش اول خود در ابتدا در یک پردازش اولیه کلمات ایست را از پیکره حذف نمودیم و سپس به آموزش مدل‌های خود پرداختیم اما در این روش این کار به اثرات حاصل از فرمول idf سپردیم که نتایج نشان می‌دهد که بهتر است حذف کلمات ایست را به عنوان بخشی از اقدامات پیش پردازشی خود انجام دهیم.

معیار	داده ترین	داده تست
NMI	0.52	0.54
Accuracy	0.66	0.67
F-Measure	0.64	0.66
V-Measure	0.52	0.54

روش سوم (doc2vec):

در این روش برای بازنمایی هر سند از الگوریتم doc2vec استفاده شد. این روش در واقع یک الگوریتم بدون سرپرستی می‌باشد که در آن برای جمله/پاراگراف/داکیومنت یک بردار ایجاد می‌کند. این الگوریتم در واقع الهام گرفته از word2vec می‌باشد که در آن برای هر کلمه یک بردار ایجاد می‌کند. در این روش کلمات به صورت مستقل از هم در نظر گرفته نمی‌شود و یک رابطه‌ی سیکونسی بین کلمات در نظر گرفته می‌شود.

معیار	داده ترین	داده تست
NMI	0.17	0.18
Accuracy	0.51	0.50
F-Measure	0.46	0.47
V-Measure	0.17	0.18

این روش نسبت به دو روش قبل دارای نتایج ضعیف‌تری می‌باشد که این موضوع کمی عجیب به نظر می‌رسد چرا که انتظار ما این بود که در این روش نسبت به دو روش قبل نتایج بهتری مشاهده کنیم. به همین دلیل به بررسی چندین باره روند کار پرداختیم اما متوجه دلایل نتایج ضعیف نشدیم.

روش چهارم (document-term):

در این روش برای بازنمایی هر سند از ماتریس سند-کلمه استفاده شد. اندازه‌ی این ماتریس برابر با مقداری در حدود 65000*8600 شد. برای بازنمایی از کل سندهای آموزش و ارزیابی استفاده شد. بعد از این که ماتریس حاصل محاسبه شد ما برای این که آن را تبدیل به یک ماتریس dense کنیم از الگوریتم svd استفاده کردیم. از آنجایی که این ماتریس دارای مقادیر صفر زیاد بود برخی از الگوریتم‌ها در زمان مناسب پاسخ‌گو نبوده و ما مجبور به استفاده از الگوریتم‌های خاصی برای بدست آوردن ماتریس dense شدیم. الگوریتم مورد استفاده ما svds که از کتابخانه‌ی scipy است بود.

بعد از بدست آوردن ماتریس dense با استفاده از قسمت مربوط به آموزش آن، مدل خوشه‌بندی خود را آموزش دادیم. بعد از خوشه‌بندی، طبق الگوی گفته شده هر خوشه را ورچسب‌گذاری کرده و به محاسبه معیارهای خواسته شده پرداختیم. در ادامه به سراغ داده تست رفته و بازنمایی که برای هر کدام از این اسناد به دست آورده بودیم را به مدل خوشه‌بند خود داده تا ورچسب آن را پیش‌بینی کند. سپس به محاسبه معیارهای خواسته شده پرداختیم که نتایج حاصل آن به صورت زیر می‌باشد:

معیار	داده ترین	داده تست
NMI	0.04	0.04
Accuracy	0.28	0.27
F-Measure	0.42	0.42
V-Measure	0.04	0.04

این روش نسبت به روش‌های قبل از نتایج ضعیف‌تری برخوردار می‌باشد که به نظر نویسنده یکی از دلایل آن می‌تواند استفاده از الگوریتم svd نامناسب باشد.