دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

پروژه یادگیری ماشین
آنالیز احساس متن

نگارش

رضا صومی

استاد

دکتر مطهری

شهریور ۱۴۰۱

چکیده

در این پروژه تجزیه و تحلیل احساسات بر روی مجموعه داده ای شامل نظرات بینندگان درباره فیلم ها را ارائه می دهد. هر نظر با یک امتیاز احساسی از ۰ (بسیار منفی) تا ۴ (بسیار مثبت) همراه است. علاوه بر این، مجموعه داده شامل عبارات جزئی هر عبارت با نمرات متناظر است که به نوعی متن را تقسیم کرده و برای هر زیربخش آن یک امتیاز در نظر می گیرد. کار تجزیه و تحلیل احساسات با استفاده از دو روش انجام می شود:

مدل FastText + شبکه عصبی: عبارات با استفاده از مدل FastText به یک فضای وکتور ۱۰۰ بعدی reduce شده اند و یک شبکه عصبی برای پیش بینی امتیازات احساسات بر اساس این embedding ها آموزش داده می شود.

مدل BERT + دسته بند آن : عبارات با استفاده از مدل BERT به فضای ۷۵۶ بعدی reduce شده اند و از دسته بندی BERT برای انجام وظیفه تجزیه و تحلیل احساسات استفاده می شود.

این مستندات پیاده سازی و ارزیابی هر دو روش را توضیح می دهد، نقاط قوت مربوطه را برجسته می کند و نتایج به دست آمده از هر رویکرد را مورد بحث قرار می دهد.

کلیدواژه‌ها: یادگیری ماشین، پردازش زبان طبیعی، آنالیز احساسات متن، مدل FastText و Bert و روش SkipGram

فهرست مطالب

فهرست شکل‌ها

فصل ۱

رویکردها و توضیحات

در ادامه لیست کتابخانه های استفاده شده و علت استفاده آنها، نحوه استخراج ویژگی از داده ها یا همان embedding عبارات و مدل های به کارگرفته شده و علت استفاده آنها و تعداد پارامتر هرکدام بررسی می شود.

۱ـ۱ کتابخانه های به کارگرفته شده

کتابخانه مورد نظر و هدف استفاده از آن در ادامه لیست شده است.

pandas: load کردن دیتاست به عنوان یک dataframe و کارکردن با آن

sklearn: استفاده از توابعی نظیر train-test-split برای تقسیم بندی داده ها، استفاده از کلاس Tfid-fVectorizer آن برای بدست آوردن خودکار tf و idf هر توکن و استفاده از توابع موجود برای ارزیابی نظیر f۱-score و precision-score

fastText: بدست آوردن embedding هر کلمه در corpus و استفاده از متود skip-Gram در آن

torch: آموزش مدل شبکه عصبی با استفاده از آن برای استفاده از GPU

Transformers: استفاده از مدل از پیش آموزش داده شده Bert جهت بدست آوردن embedding کلمات و استفاده از توابع آماده جهت اعمال classification برای تسک مورد نظر

timetqdm: مورد استفاده در فرآیند آموزش شبکه عصبی برای ثبت زمان و آموزش شبکه عضبی به صورت mini-batch

۱ـ۲ نحوه استخراج ویژگی از داده ها در رویکرد ۲

در ادامه نحوه استخراج ویژگی از داده ها توضیح داده می شود. برای این کار ابتدا از روش fasttext استفاده شده است و برای بهبود عملکرد به سراغ مدل از پیش آموزش داده شده bert رجوع می کنیم.

FastText (Skip-gram): FastText الگوریتمی برای یادگیری جاسازی کلمات است که کلمات را به صورت نمایش برداری پیوسته در فضایی با ابعاد بالا نشان می دهد. Skip-gram یکی از روش هایی است که FastText برای ایجاد جاسازی کلمات استفاده می کند. در اینجا نحوه کار آن آمده است: برای هر کلمه یا زیرکلمه، FastText جفت های context-center ایجاد می کند. زمینه، کلمات بافت اطراف است و هدف، کلمه مرکزی یا زیرکلمه است. به عنوان مثال، با توجه به جمله "من عاشق سیب هستم"، یک جفت می تواند این باشد که عاشق کلمه مرکزی باشد و سیب و من کلمات کناری باشد. و با توجه به ساختار آن این مدل برای هر کلمه دو وکتور u و v بدست می آورد و وکتور نهایی را از جمع این دو وکتور برای هر کلمه می سازد. به عبارتی در آموزش یک شبکه عصبی را با استفاده از این جفت ها آموزش می دهد تا کلمه یا زیرکلمه مورد نظر را با توجه به بافت آن پیش بینی کند. این فرآیند نمایش برداری کلمات یا زیرکلمه ها را می آموزد. جاسازی‌های کلمه‌ای که به‌دست می‌آیند، اطلاعات معنایی و نحوی را جمع‌آوری می‌کنند و به کلماتی با زمینه‌های مشابه اجازه می‌دهند تا بازنمایی‌های برداری مشابهی داشته باشند.

BERT: BERT یک مدل زبان مبتنی بر ترانسفورماتور است که تعبیه‌های متنی کلمه را ایجاد می‌کند. برخلاف تعبیه‌های سنتی کلمه، BERT کل بافت یک کلمه را به جای صرفاً بافت محلی در نظر می‌گیرد. در اینجا یک نمای کلی از نحوه عملکرد BERT توضیح داده می شود: پیش آموزش: BERT بر روی مجموعه بزرگی از متن با استفاده از یک هدف مدل سازی زبان پوشانده شده از قبل آموزش داده شده است. در حین پیش تمرین، BERT یاد می گیرد که کلمات پوشیده شده را در یک جمله پیش بینی کند. این فرآیند BERT را قادر می‌سازد تا اطلاعات متنی را جمع‌آوری کند و جاسازی‌هایی را ایجاد کند که از کلمات اطراف آگاه هستند. تنظیم دقیق: پس از پیش‌آموزش، BERT را می‌توان برای کارهای پایین دستی خاص، مانند تجزیه و تحلیل احساسات، به‌خوبی تنظیم کرد. مدل BERT پیش آموزش دیده بیشتر بر روی داده های خاص کار آموزش داده می شود تا جاسازی ها را برای کار خاص تطبیق دهد. این تنظیم دقیق به BERT اجازه می دهد تا در وظایف مختلف پردازش زبان طبیعی به خوبی عمل کند. تعبیه‌های متنی تولید شده توسط BERT قادر به ثبت الگوهای زبانی پیچیده هستند و می‌توانند عملکرد وظایفی مانند تجزیه و تحلیل احساسات را با استفاده از اطلاعات زمینه بهبود بخشند.

به طور خلاصه، FastText (Skip-gram) جاسازی‌های کلمه‌ای را بر اساس زمینه و زیرکلمه‌های محلی ایجاد می‌کند ، در حالی که BERT با در نظر گرفتن کل متن جمله، جاسازی‌های کلمه‌ای را ایجاد

می‌کند. هر دو روش نمایش قدرتمندی برای کلمات ارائه می دهند و می توانند به طور موثر در وظایف مختلف پردازش زبان طبیعی استفاده شوند. همچنین لازم به ذکر است مدل fasttext روی داده های مورد نظرمان آموزش داده می شود و برای هر کلمه یک جاسازی بدست می آورد، اما در bert تنها کافیست از tokenizer آن استفاده کرده و با ورودی دلخواه خروجی مورد نظر را بدون آموزش روی دیتاست موجود بدست آوریم. همچنین استفاده از رویکرد ۱ به جهت دقت کمتر و کار اضافه بیشتر به این کتابخانه ها مهیا می کنند، صرف نظر شده است.

حال پس از پیداکردن embedding برای هر کلمه، embedding هر عبارت را برابر میانگین بردار جاسازی کلمات موجود در عبارت در نظر می گیریم. لازم به ذکر است برای مدل bert کافیست از cls token که نماینده آن عبارت است استفاده کنیم.

۱ـ۳ مدل ها، اهداف به کارگیری و تعداد پارامترهای هر کدام

پس از استخراج بردار جاسازی ۱۰۰ بعدی در fasttext یا ۷۵۶ بعدی در مدل Bert کافیست این را به عنوان ورودی به مدل یادگیری ماشین مورد نظرمان دهیم. برای این کار از قسمت Deep Learning تدریس شده استفاده می شود و یک شبکه عمیق که از ۴ لایه میانی تشکیل شده است، استفاده می شود. همچنین از تابع فعالساز ReLU برای ایجاد پارامتر غیر خطی مدل استفاده می شود. استفاده از مدل شبکه عصبی به این دلیل است که پارامتر های ورودی و دیتای مورد نظر به گونه ای است که از سایر تکنیک ها نظیر توابع خطی یا ensamble method ها نمی توان استفاده کرد. به عبارتی در صورت استفاده از متودهایی نظیر درخت تصمیم به نتایج بسیار ناخوشایندی دچار می شویم. (که از موارد تست شده به شمار می رود) در صورتی که یادگیری عمیق تعمیم خوبی را به همراه می اورد. در نتیجه ورودی شبکه عصبی مورد نظر embedding ها بوده و خروجی آن یکی از ۵ کلاس مورد نظر برای تحلیل احساسات متن خواهد بود. همچنین تعداد پارامتر های شبکه عصبی طراحی شده ۲۳۲۰۰ پارامتر است. برای قسمت Bert نیز از BertSequenceClassification که توسط hugging face طراحی شده است، استفاده شده است و پارامتر های تعداد پارامترهای bert نیز ۱۱۰ میلیون است. لازم به ذکر است از uncased based bert استفاده شده است که پارامترهای کمتری دارد.

فصل ۲

ارزیابی و تحلیل

۲ـ۱ مقدمه

داده های موجود برای آموزش نامتعادل هستند. به عبارتی کلاس ۲ یا همان خنثی اکثریت داده موجود را تشکیل می دهد و لذا باعث می شود مدل تعمیم بهتری برای این کلاس داشته باشد و کلاس های دیگر را به نسبت خوب تشخیص ندهد. برای جلوگیری از این کار یک undersampling برای این کلاس و یک oversampling برای کلاس با داده کمتر نظیر کلاس ۴ انجام می دهیم تا داده ها متعادل شوند. این کار بسیار مهم است و نتیجه موثری در خروجی کار دارد. لذا با پرداختن به عدم تعادل کلاس و انجام مهندسی ویژگی موثر، مدل می‌تواند الگوها را بهتر ثبت کند و پیش‌بینی‌های دقیقی را در همه کلاس‌ها انجام دهد. اما در صورتی که از این روش استفاده شود، دقت مدل نهایی به شدت کاسته می شود چرا که در دنیای واقعی داده داده ها مانند همین دیتاست unbiased هستند و الزام است که داده های مهم که بیشتر تکرار شده اند را مدل بهتر بفهمد و تعمیم خوبی برای آنها داشته باشد. لذا در تسک خاص ما از این کار صرف نظر شده است تا نتیجه بهتری حاصل شود. همچنین در رابطه با preprocessing دیده شده که مدل هایی نظیر bert در صورتی که پیش پردازش انجام نگیرد، دقت بهتری دارند و کلمات و punctuation ها مهم هستند. لذا در این تسک از این کار نیز صرف نظر شد اما کد آن موجود می باشد.

۲ـ۲ نتایج ارزیابی مدل ها و تحلیل نقاط قوت و ضعف مدل

در فایل پیوست ژوپیتر نوتبوک می توانید نتایج را مشاهده کنید. در اینجا توضیحاتی داده می شود. روش اول توضیح داده شده یا به عبارتی استفاده از fastText و شبکه عصبی دقت یا همان accuracy که برابر

micro-f۱ را ۶۰ درصد برآورد کرد. حال آنکه نتیجه بهتری از مدل شبکه عصبی انتظار می رود. دلیل آن می تواند چندین پارامتر باشد. یکی از دلایل آن می تواند این باشد که مدل fastText به خوبی بردار جاسازی را نیافته است چرا که هنگامی که از روش دوم یا bert استفاده می کنیم با اینکه تعداد پارامترها بیش از ۱۰۰۰ برابر روش اول است و آپدیت وزن ها به نظر می رسد زمان زیادی را بگیرد اما در زمانی تقریبا یکسان نتیجه افزایش حدود ۱۰ درصدی روش دوم و برآورد دقت ۷۰ درصدی است. به نظر می رسد با توجه به اینکه داده های موجود هم نوع نیستند و یک عبارت تقریبا ۲۰ زیرعبارت دارد که هرکدام نیز درون دیتاست هستند (حتی یک کلمه درون عبارت نیز یک عبارت مجزا به حساب می آید) می توان گفت با اعمال مهندسی ویژگی کمی پیشرفته تر و در نظر گرفتن موارد متعدد می توان تعمیم بهتری حتی با استفاده از همین مدل نیز داشت اما با توجه به اینکه محدودیتی وجود ندارد و می توان از هر کتابخانه ای استفاده کرد روش دوم به عنوان روش برتر شناخته می شود و نتیجه مناسبی روی داده تست دارد. از نقاط قوت روش اول می توان به این مورد اشاره کرد که سریع تر آموزش صورت می گیرد، فضای کمتری را اشغال می کند، برای محیط هایی که تنها از کلمات محدودی استفاده می شود موثر است و روش دوم نقاط قوت آن آموزش بهتر و تعمیم درست تر و دقت بالاتر است. نقاط ضعف نیز پیش از آن در قسمت اول توضیح داده شد.

در شکل ۲ـ۱ هزینه در هر epoch برای داده های train و validation و همچنین accuracy را مشاده می کنید. لازم به دکر است با کاهش هزینه training مشاهده می شود که accuracy ثابت باقی می ماند. در این حالت می توان گفت مدل بیش از این قادر به کشف محیط نبوده و درک مدل بیش از آن نخواهد بود. همچنین نیاز به epoch های چهارم و پنجم نیست و با ۳ دور اول به دقت مورد قبول می رسیم.

Epoch	Training Loss	Validation Loss	Accuracy
1	0.750100	0.739232	0.687601

شکل ۲ـ۱: خروجی مدل دسته بندی BERT

۲ـ۳ تلاش های شکست خورده

استفاده از مدل های خطی واضحا جوابگو نخواهد بود چرا که مدل در پی تخمین ۵ کلاس به عنوان خروجی خواهد بود و مدل های خطی نمی توانند این کار را انجام دهند. لازم به ذکر است از svm با کرنل غیر خطی می توان استفاده کرد اما در اینجا پاسخ مطلوبی بدست نمی اید چرا که تعداد ویژگی ها هم در روش اول

و هم روش دوم زیاد است و svm نمی تواند تخمین خوبی در این شرایط مهیا کند. همچنین SVM های غیر خطی در درجه اول زمانی موثر هستند که مرز تصمیم گیری بین کلاس ها بسیار پیچیده یا غیرخطی باشد. با این حال، تجزیه و تحلیل احساسات لزوماً بر مرزهای تصمیم گیری پیچیده متکی نیست، بلکه بیشتر بر گرفتن اطلاعات معنایی و متنی در متن است.
علاوه بر آن استفاده از روش های learning ensamble نیز دقت مورد قبولی در این تسک ندارد. یکی از دلایل آن همانند مورد قبلی فضای ویژگی بالا است. همچنین تحلیل احساسات به شدت بر درک زمینه‌ای تکیه می‌کند و معنا و احساسات پشت کلمات یا عبارات را به تصویر می‌کشد. روش‌های مجموعه‌ای مانند بسته‌بندی و تقویت معمولاً زمانی به خوبی کار می‌کنند که مدل‌های فردی قادر به گرفتن جنبه‌ها یا تغییرات مختلف داده‌ها باشند. با این حال، این روش‌ها ممکن است برای به دست آوردن موثر زمینه ظریف مورد نیاز برای تجزیه و تحلیل احساسات با مشکل مواجه شوند، زیرا آنها عمدتاً بر ترکیب چندین مدل بدون در نظر گرفتن تفاوت‌های ظریف زبان تمرکز می‌کنند. لازم به ذکر است درخت‌های تصمیم که معمولاً در روش‌های گروهی استفاده می‌شوند، معمولاً بر اساس ویژگی‌های فردی و آستانه‌های آنها تصمیم می‌گیرند. آنها ذاتاً اطلاعات متوالی یا وابستگی بین کلمات یک جمله را نمی گیرند. در تجزیه و تحلیل احساسات، ترتیب و چینش کلمات می تواند به طور قابل توجهی بر احساس بیان شده تأثیر بگذارد. بنابراین، درخت های تصمیم ممکن است نتوانند به طور کامل از ماهیت متوالی داده های متنی استفاده کنند و عملکرد آنها را محدود کنند.

شبکه‌های عصبی و یادگیری عمیق: تحلیل احساسات از پیشرفت‌های شبکه‌های عصبی و مدل‌های یادگیری عمیق مانند FastText و BERT بسیار سود برده است. این مدل‌ها در ثبت الگوهای پیچیده زبانی، اطلاعات زمینه‌ای و وابستگی‌های بلندمدت در متن عالی هستند. آنها عملکرد برتر در وظایف مختلف پردازش زبان طبیعی، از جمله تجزیه و تحلیل احساسات، نشان داده اند. بنابراین، استفاده از این مدل‌های مبتنی بر شبکه عصبی ممکن است نتایج بهتری در مقایسه با روش‌های مجموعه سنتی داشته باشد.

۲ـ۴ مقایسه رویکردهای اولیه و نوین

در این قسمت مقایسه بین روش سنتی شامل preprocessing روی متون و بدست آوردن بردار tf-idf از این طریق و روشی که استفاده شد، را بررسی می کنیم.

بیایید این رویکردها را با هم مقایسه کنیم:

پیش پردازش + TF-IDF: روش پیش پردازش و TF-IDF یک روش سنتی و مبتنی بر قانون برای نمایش متن است. این شامل چندین مرحله برای تمیز کردن و عادی سازی داده های متنی قبل از تولید

بردارهای ویژگی با استفاده از TF-IDF است. در اینجا به برخی از ویژگی های این رویکرد اشاره می شود:
مزایا:

سادگی: مراحل پیش پردازش ساده هستند و به راحتی قابل اجرا هستند.
تفسیرپذیری: بردارهای TF-IDF نمایش ساده ای از اهمیت اصطلاحات در متن ارائه می دهند.
معایب:

درک متنی محدود: این روش ذاتاً اطلاعات متنی و وابستگی بین کلمات در متن را در بر نمی گیرد. با هر کلمه به طور مستقل برخورد می کند.
عدم درک معنایی: ریشه کردن و حذف کلمات توقف ممکن است منجر به از دست رفتن اطلاعات معنایی شود که به طور بالقوه بر عملکرد وظایفی مانند تجزیه و تحلیل احساسات تأثیر می گذارد.
قدرت بازنمایی ناکافی: نمایش های TF-IDF ممکن است الگوهای پیچیده زبانی و تفاوت های ظریف مورد نیاز برای تجزیه و تحلیل احساسات دقیق را دربر نگیرد. ممکن است برای مدیریت کلمات خارج از واژگان یا اصطلاحات نادر مشکل باشد.
FastText و BERT: FastText و BERT مدل های پیشرفته مبتنی بر شبکه عصبی هستند که به صراحت برای وظایف پردازش زبان طبیعی طراحی شده اند. آنها مزایای قابل توجهی نسبت به روش های سنتی مانند پیش پردازش + TF-IDF دارند:
درک متنی: FastText و BERT با در نظر گرفتن کلمات و جملات اطراف، اطلاعات متنی را ضبط می کنند. این به آنها امکان می دهد معنی و احساسات پشت کلمات را با دقت بیشتری درک کنند.

درک معنایی: بر خلاف ریشه و حذف کلمه توقف، مدل‌های FastText و BERT اطلاعات معنایی را حفظ می‌کنند و آنها را قادر می‌سازد تا عبارات ظریف‌تری را در متن ثبت کنند.

قدرت بازنمایی: مدل‌های FastText و BERT از تکنیک‌های یادگیری عمیق برای یادگیری الگوهای پیچیده، وابستگی‌های دوربرد و نشانه‌های زبانی ظریف استفاده می‌کنند. آنها می توانند کلمات خارج از واژگان را مدیریت کنند و در مقایسه با روش های سنتی قدرت بازنمایی بالاتری دارند.

عملکرد پیشرفته: مدل‌های FastText و BERT در کارهای مختلف پردازش زبان طبیعی، از جمله تجزیه و تحلیل احساسات، به عملکرد پیشرفته‌ای دست یافته‌اند. آنها بر روی مجموعه های گسترده آموزش دیده اند و می توانند از دانش آموخته شده خود برای پیش بینی دقیق استفاده کنند.

به طور خلاصه، در حالی که رویکرد پیش پردازش + TF-IDF ساده تر و قابل تفسیرتر است، ممکن است فاقد درک متنی و معنایی مورد نیاز برای تجزیه و تحلیل احساسات دقیق باشد. از سوی دیگر، مدل‌های FastText و BERT در گرفتن الگوهای پیچیده زبانی، اطلاعات زمینه‌ای و تفاوت‌های معنایی

برتری دارند و آنها را برای وظایف تحلیل احساسات مناسب‌تر و مؤثرتر می‌سازد.