سیستم های اطلاعاتی هوشمند
نشریه علمی دانشگاه آزاد اسلامی
مجلد 3 ، شماره 6
صفحات 36-21

یادگیری عمیق در تحلیل احساسات کاربران شبکه های هوشمند سرمایه گذاری

نسرین صالحی چگنی[۱]، صبا جودکی[۲*]، مجتبی صالحی[۳]

*نویسنده مسئول: دریافت: 1403/7/16 ، بازنگری: 1403/9/20 ، پذیرش: 1403/11/19

[۱]گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران
[۲*]گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران (saba.joudaki@iau.ac.ir)
[۳]گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران

چکیده

تحلیل داده‌های سهام شرکت ها یکی از روش‌های مهم برای ارزیابی عملکرد و ارزش شرکت‌ها و انتخاب بهترین گزینه برای سرمایه‌گذاری در بازارهای مالی است. در این مقاله، مجموعه داده نظرات کاربران شبکه هوشمند سرمایه‌گذاری سهام یاب با هدف تحلیل احساسات کاربران گردآوری شده است. ابتدا الگوریتم‌های درخت تصمیم، ماشین بردار پشتیبان، بیزین ساده و نزدیکترین همسایه پیاده‌سازی شدند؛ که بردار پشتیبان با صحت ۶۱ درصد بهترین عملکرد را به‌دست آورد. سپس برای مقایسه این الگوریتم‌های یادگیری سنتی با الگوریتم‌های یادگیری عمیق، شبکه‌های LSTM و BERT به زبان فارسی و BERT به زبان انگلیسی پیاده سازی شدند. این مدل ها به ترتیب با صحت های ۷۲ ، ۸۲ و ۸۳ درصد عملکرد بهتری نسبت به الگوریتم های سنتی داشتند. در ادامه مدل LSTM با استفاده از الگوریتم فرا ابتکاری ژنتیک با هدف به دست آوردن هایپر پارامترهای بهینه پیاده سازی شد که به صحتی برابر با ۸۱/۴۶ درصد رسید. در فاز پایانی، الگوریتم BRET دو زبانه با ترکیب متن فارسی نظرات و معنای انگلیسی آن‌ها پیاده سازی شد و به عملکرد ۸۴ درصد دست یافت. انتظار می‌رود استفاده از این مدل بتواند به بهبود عملکرد سیستم‌های پیش‌بینی کننده و توصیه گر در سایت‌های اقتصادی کمک نماید.

کلمات کلیدی: پردازش زبان طبیعی، تحلیل احساسات، یادگیری عمیق، شبکه عصبی، سیستم‌های توصیه‌گر

۱- مقدمه

تجزیه و تحلیل احساسات یا نظر کاوی یک حوزه در حال پیشرفت است که به‌استفاده از پردازش زبان طبیعی، تجزیه و تحلیل متن اشاره دارد و برای استخراج کمیت مورد استفاده قرار می‌گیرد و برای مطالعه حالات احساسی از یک بخش مشخص از اطلاعات یا مجموعه داده‌های متنی مورد استفاده قرار می‌گیرد. تحلیل احساسات در بسیاری از شرکت‌ها برای بررسی محصولات، نظرات رسانه‌های اجتماعی و برای بررسی مثبت، منفی یا خنثی بودن متن استفاده می‌شود. بیشتر مطالعات در زمینه تحلیل احساسات در زبان انگلیسی از روش‌های یادگیری ماشین (Machine Learning) سنتی مانند SVM، درخت تصمیم‌گیری و Naïve Bayes استفاده می‌کنند. این روش‌ها معمولاً از ویژگیهای n-gram و لغوی بهره می‌برند. در زبان‌شناسی مفهوم n-gram به معنای دنباله‌ای پیوسته از n جزء در یک دنباله معین از متون است. این اجزا می‌توانند حروف، واج، هجا یا واژه باشند.

از سویی دیگر، روش های سنتی در نادیده گرفتن اطلاعات معنایی مندرج در متن محدودیت دارند. یکی دیگر از مسایل مطرح در روش‌های سنتی یادگیری ماشین مبحث مهندسی ویژگی است که دارای روالی طولانی و سخت است. برای حل این مشکلات و دستیابی به نتایج بهینه، استفاده از روش‌های یادگیری عمیق (Deep Learning) نسبت به روش‌های سنتی یادگیری ماشین از اولویت بالاتری برخوردار می‌باشد. یادگیری عمیق شاخه‌ای از یادگیری ماشین است. به عبارت دیگر همان یادگیری به وسیله شبکه‌های عصبی است که لایه‌های مخفی (Hidden Layers) زیادی دارد. روش‌های یادگیری ماشین در دو دسته اصلی با ناظر در صورت وجود برچسب و بدون ناظر در حالت بدون برچسب تقسیم می‌شوند که بیش‌تر کارهای انجام شده در سطح تحلیل متن مبتنی بر روش‌های با ناظر می‌باشند.

بازارمالی به عنوان یک سیستم پیچیده با نوسانات غیرخطی از جمله بخش‌هایی است که از مزایای الگوریتم‌های متن کاوی بهره برده است. به عنوان مثال، متن کاوی در زمینه های مختلفی مانند درک و مدیریت ریسک مالی، مدیریت تسهیلات، رتبه بندی اعتباری مشتریان، تجزیه و تحلیل و رتبه بندی مشتریان بانکی، پیش

بینی سود سهام، پیش بینی بازار بورس، پیش بینی نرخ ارز، پیش بینی ورشکستگی بانک‌ها، تحلیل احساسات سرمایه‌گذاران، تحلیل نظرات مشتریان و تشخیص کلاه برداران مالی به ایفای نقش پرداخته است. عموماً سرمایه‌گذاران بازار مالی به منظور سود بیش‌تر، نظرات و حالت‌های احساسی خود را با دیگر سرمایه‌گذاران به اشتراک می‌گذارند و در نتیجه رفتار سرمایه‌گذاران مالی بر قیمت‌های بازار سهام تاثیرگذار خواهد بود. ولی از آن جایی‌که غالب سرمایه‌گذاران از روند تغییرات بازارهای مالی اطلاع ندارند، بررسی حالت‌های احساسی سرمایه‌گذاران و شناسایی الگوهای مخفی بین نظرات مردم در رسانه‌های اجتماعی، می‌تواند آن‌ها را برای رسیدن به سودآوری بیش‌تر، شناخت تغییرات بازار و پیش‌گویی حوادث اقتصادی یاری کند. امروزه محققان در صدد هستند تا با به‌کارگیری تکنیک‌های هوش مصنوعی مانند شبکه‌های عصبی پیچشی و یا سایر نمونه‌های شبکه‌های عصبی عمیق به ارزیابی نظرات، تحلیل داده‌های احساسی مشتریان یا کاربران در فضاهای مجازی و نیز شناسایی و پیش‌بینی تغییرات در بازار بپردازند. سیستم‌های هوشمند تحلیل‌گر احساسات، نقش مهمی را در تحلیل و طبقه‌بندی نظرات، توصیف ها و نگرش‌های مردم نسبت به موضوعات مطرح در کامنت‌های شبکه‌های اجتماعی بازی می‌کنند.

هدف این مقاله ارائه یک مدل مبتنی بر شبکه عصبی عمیق به‌منظور تحلیل نظرات کاربران سایت سهامیاب و مقایسه دقت و کارایی مدل پیشنهادی با مدل‌های سنتی یادگیری ماشین است. برای این منظور، نظرات کاربران سایت سهامیاب در خصوص بورس تهران جمع‌آوری شد. سپس بار معنایی جملات با استفاده از تکنیک‌های متن کاوی و آنالیز احساسات تعیین گردید و با استفاده از الگوریتم‌های یادگیری ماشینی به دسته‌های مثبت و منفی طبقه‌بندی شد به این شکل که نظرات مثبت با عدد ۱ و نظرات منفی با عدد صفر برچسب گذاری شدند.

از آن‌جا که متن، داده‌ای بدون ساختار است، پیش‌پردازش برای تبدیل این داده‌های بدون ساختار به فرم ساختار یافته مورد نیاز است. سپس با انجام فرآیند یادگیری با استفاده از داده‌های مجموعه آموزشی (Train) الگوریتم های ماشین یادگیر پیاده‌سازی شده و مدل یادگیری ماشین ساخته می‌شود و درنهایت با استفاده از داده‌های آزمایشی (Test) ارزیابی می‌شود. در حالت کلی مراحل طراحی و پیاده‌سازی به صورت جمع‌آوری مجموعه داده، برچسب‌گذاری داده‌ها، پیش پردازش داده، تشکیل بردار کلمه، اجرای الگوریتم های سنتی یادگیری ماشین، طراحی شبکه عمیق، اجرای الگوریتم ژنتیک، ارزیابی کلی نتایج و ارائه مدل نهایی بیان می‌شود.

۲ - مطالعات انجام شده

بستان و همکاران [۱] درون سازی واژگان فارسی با استفاده از الگوریتمBERT را مورد بررسی قرار دادند و به درک معنایی هر واژه بر مبنای بافت متن پرداختند. مدل ایجاد شده بر روی مجموعه دادگان وب فارسی مورد پیش آموزش قرارگرفت و پس از طی دو مرحله تنظیم دقیق با معماری های متفاوت، مدل نهایی تولید شد. نتایج حاصل از این مدل بهبود خوبی نسبت به سایر مدل‌های مورد بررسی داشت و دقت را نسبت به مدل BERT چندزبانه تاحداقل یک درصد افزایش داد.

انتونا کاکی و همکاران [۲] موضوعات تحقیقاتی فعلی را در توییتر با تمرکز بر سه حوزه اصلی ترسیم کردند: ساختار و ویژگی‌های نمودار اجتماعی، تحلیل احساسی و تهدیدهایی مانند اسپم، ربات‌ها، اخبار جعلی و سخنان نفرت آمیز. همچنین مدل داده پایه توییتر و بهترین روش‌ها برای نمونه‌گیری و دسترسی به داده را ارائه دادند و زمینه تکنیک‌های محاسباتی مورد استفاده در این حوزه‌ها مانند نمونه‌برداری گراف، پردازش زبان طبیعی و یادگیری ماشینی را ارائه دادند.

بصیری و همکاران [۳] اولین مدل عمیق دو جهتی CNN - RNN مبتنی بر توجه را برای تحلیل احساسی پیشنهاد کردند.. مدل آن‌ها هم در طبقه‌بندی طولانی و هم در طبقه‌بندی کوتاه قطبیت توییت به نتایج سطح بالا دست یافت. هم‌چنین، مکانیزم توجه بر روی خروجی‌های لایه‌های دو طرفه ABCDM برای تأکید بیش‌تر یا کم‌تر بر روی کلمات مختلف اعمال می‌شود. برای کاهش ابعاد ویژگی‌ها و استخراج ویژگی‌های محلی مقاوم در برابر موقعیت، ABCDM از مکانیزم‌های کانولوشن و ادغام استفاده می‌کند. اثربخشی ABCDM بر تشخیص قطبیت احساسی که رایج‌ترین و اساسی‌ترین وظیفه تحلیل احساسی است، ارزیابی می‌شود. آزمایش‌ها در پنج بررسی و سه مجموعه داده توییتر انجام شده‌است. نتایج مقایسه ABCDM با شش DNN که اخیرا برای تجزیه و تحلیل احساسی پیشنهاد شده‌اند نشان می‌دهد که ABCDM به نتایج بسیار خوبی در بررسی بلند مدت و طبقه‌بندی قطبیت کوتاه توییت دست می‌یابد.

دشتی پور و همکاران [۴] یک مجموعه داده چند وجهی فارسی شامل بیش از ۸۰۰ جمله را به‌عنوان یک منبع معیار برای محققان به منظور ارزیابی رویکردهای تحلیل احساسات چند وجهی در زبان فارسی ارائه دادند. یک چارچوب جدید تحلیل احساسات چند وجهی آگاه از زمینه را که به طور همزمان از نشانه‌های صوتی، بصری و متنی برای تعیین دقیق‌تر احساسات بیان‌شده بهره می‌برد. از هر دو روش ترکیب سطح تصمیم (دیرهنگام) و سطح ویژگی (زود هنگام) برای ادغام اطلاعات موثر متقابل استفاده کردند. نتایج تجربی نشان می‌دهند که ادغام بافتی ویژگی‌های چند وجهی مانند ویژگی‌های متنی، صوتی و بصری عملکرد بهتری (۹۱.۳۹ ٪) را در مقایسه با ویژگی‌های تک وجهی (۸۹.۲۴ ٪) ارائه می‌دهند.

آک و همکاران [۵] از اخبار مالی برای پیش‌بینی عملکرد بازار سهام استفاده کردند. هنگ [۶] یک سیستم پیش‌بینی را براساس روش متن کاوی و اخبار بازار سهام پیشنهاد کرد. برای پاسخ به تغییرات بازار سهام در زمان واقعی، از LSTM استفاده کرد و براساس داده‌های تحلیل سری زمانی گذشته، نزدیک‌ترین وضعیت به زمانی را یافت که قیمت سهام با محاسبات ریاضی افزایش یافت.

لی و همکاران [۷]۴۳۰۰ نظر را با احساسات بسیار منفی / مثبت منتشر شده در وب سایت‌های همسریابی به‌عنوان یک نمونه انتخاب کردند و هنگام تست و مقایسه کارایی تحقیق رفتار کاربر، از تحلیل احساسی مختلف، تکنیک‌های یادگیری ماشین و تحلیل احساسی مبتنی بر فرهنگ لغت استفاده کردند ودریافتند که ترکیب یادگیری ماشین و روش مبتنی بر واژگان می‌تواند به‌دقت بالاتری نسبت به هر نوع تحلیل احساسی دست یابد.

در تحقیقی دیگر با ایجاد سیستم‌های جدید تحلیل‌گر احساسات بر مبنای تکنیک‌های مختلفی از شبکه عصبی عمیق، دقت پیش‌بینی سهام بورس را بهبود داده شد تا سرمایه گذاران بورس بتوانند با آگاهی بهتر از روند تغییرات بورس، بازده سهام خود را افزایش دهند [۹-۸] .

لوتز و همکاران [۱۰] یک رویکرد یادگیری ماشینی جدید را برای پیش‌بینی برچسب‌های قطبیت سطح جمله در اخبار مالی توسعه دادند. روش آن‌ها از نمایش متن توزیع شده و یادگیری چند نمونه‌ای برای انتقال اطلاعات از سطح سند به سطح جمله استفاده کرد. سیستم خبره پیشنهادی می‌تواند به سرمایه‌گذاران در تصمیم‌گیری خود کمک کند و ممکن است به آن‌ها در برقراری ارتباط با پیام‌های مورد نظر کمک کند.

رووییمن و همکاران [۱۱] یک الگوریتم تجزیه و تحلیل احساسی متن چینی را بر اساس BERT و CNN پیشنهاد می‌کند. این شبکه از "BERT" برای استخراج ویژگی‌های هر کلمه و استفاده از آن به عنوان ورودی سی ان استفاده می‌کند. آزمایش‌ها نشان می‌دهند که این مدل از نظر اثربخشی امکان پذیر است. مدل BERT ساختار Encode - Decoder را براساس چارچوب ترانسفورمر پیاده‌سازی می‌کند. ساختار ترانسفورمر در ترانسفورماتور هم‌چنین از چارچوب Encoder - Decoder استفاده می‌کند که ماژول رمزگذار آن از ۶ رمزگذار و ماژول رمزگشا از ۶ رمزگشا تشکیل شده است. هر کدگذار شامل یک لایه خود - توجهی و یک شبکه عصبی پیش‌خور است. توجه به خود به گره فعلی اجازه می‌دهد تا نه تنها بر روی کلمه فعلی تمرکز کند، بلکه معانی متن را نیز درک کند. هر گره Decoder شامل یک لایه خود - توجهی، یک لایه توجه و یک شبکه عصبی پیش‌خور است. لایه

توجه می‌تواند به گره فعلی در به دست آوردن محتوای کلیدی فعلی که باید به آن توجه شود، کمک کند.

میت و همکاران [۱۲] از تحلیل احساسی در اخبار بازار سهام برای پیش‌بینی استفاده کرد. آن‌ها هم‌چنین از خروجی تحلیل احساسی در الگوریتم های یادگیری ماشین برای تجزیه و تحلیل قیمت سهام استفاده کردند. مشابه این تحقیقات که روی تحلیل احساسات کار کرده اند در منابع [۱۴-۱۳] نیز به چشم می خورد.

صدر و همکاران [۱۵] لایه ادغام در شبکه پیچشی را با شبکه عصبی برگشتی جایگزین کردند تا بتوانند وابستگی‌های طولانی مدت را استخراج کرده و زیان اطلاعات محلی را کاهش دهند. آن‌ها با پیشنهاد یک شبکه ژرف چند نمایی از ویژگی های میانی استخراج شده از شبکه های عصبی پیچشی و برگشتی برای انجام طبقه بندی استفاده می‌کنند. آن‌ها اذعان داشتند که شبکه‌های عصبی ژرف مختلف به‌دلیل ساختارهای متمایز قادر به استخراج انواع مختلف ویژگی‌ها هستند، بنابراین در مدل پیشنهادی، ویژگی‌های استخراج شده از شبکه‌های عصبی ناهمگن را با استفاده از طبقه بندی کننده‌های چندنمایه ترکیب کردند تا با در نظر گرفتن ارتباط بین آن‌ها، بتوانند عملکرد تحلیل‌گر احساسات در سطح متنی را بهبود دهند.

سلام و همکاران [۱۶] فیلترینگ مشارکتی را براساس تحلیل احساسی با استفاده از یک مجموعه داده عربی برای ارائه توصیه‌هایی برای کتاب‌ها پیشنهاد کرده‌است. روش پیشنهادی دقت سیستم توصیه عربی را بهبود بخشیده و میانگین مقادیر خطا را از نظر RMSE و MAE به ترتیب به ۰.۵۵۸۳ و ۰.۱۵۵۸ کاهش داده‌است.

در تحقیقات دیگری تحلیل احساسات بیماران [۱۷] و نیز مشاهده کنندگان فیلم ها [۱۸] در سه مدل یادگیری عمیق حافظه کوتاه مدت طولانی (LSTM)، شبکه‌های عصبی عمیق (DNN) و شبکه‌های عصبی کانولوشن (CNN) بررسی شده اند و نتیجه نهایی این بوده است که این روش ها مکمل یکدیگر هستند و چه بسا انتظار می رود که از ترکیب این روش ها نتایج بهتر و جامع تری به دست آید.

۳- روش پژوهش

در این بخش مراحل روش پیشنهادی به تفکیک بیان می شود.

۳-۱- آماده سازی داده‌ها

قدم اول در پیاده سازی مدل پیشنهادی آماده سازی داده ها است. آماده کردن داده‌های ورودی یکی از مهم‌ترین مراحل پیاده سازی الگوریتم های یادگیری ماشین است. ابتدا برچسب گذاری نظرات جمع آوری شده از سایت سهامیاب (www.sahamyab.com) انجام می‌شود. سپس معادل انگلیسی نظرات با استفاده از ابزار مترجم گوگل ایجاد می‌گردد. در آخرین مرحله نظرات انگلیسی و فارسی و برچسب‌های متناظر با هرکدام از این نظرات در یک فایل اکسل با نام sahamyab_dataset ذخیره می‌شوند.

معمولا برای کار با داده‌هایی که در یک فایل اکسل ذخیره شده‌اند به کتابخانه pandas نیاز است. خوشبختانه، پانداس متدهای زیادی را فراهم می‌کند که می‌توان از آن‌ها برای بارگذاری داده‌ها از چنین منابعی در دیتافریم پانداس استفاده کرد. پانداس از تابع read_excel نیز استفاده می‌کند که می‌تواند برای خواندن داده‌های Excel در یک دیتافریم پانداس استفاده شود.

مجموعه داده‌ای که در این پژوهش استفاده می‌شود شامل نظرات کاربران سایت سهامیاب در خصوص سهام چند شرکت فعال بورس تهران است که از وب سایت سهامیاب جمع‌آوری شده‌اند. در جدول ۱ نمای کلی این مجموعه داده آمده است.

جدول ۱- نمای کلی مجموعه داده

Stock_name	Username	Review_text	Lable
نام سهام	نام کاربری	متن نظرات	۰یا ۱

۳-۲- برچسب‌گذاری مجموعه داده‌ها

برچسب‌گذاری مجموعه داده فرآیندی در یادگیری ماشینی است که در آن داده‌های خام مانند تصاویر، فایل‌های متنی، ویدئوها و غیره را می‌توان شناسایی کرد و برای ارائه زمینه‌ای که اجازه می‌دهد یک یا چند برچسب معنی‌دار و آموزنده را اضافه کرد، استفاده می‌شود. به طوری که مدل یادگیری ماشینی بتواند چیزی از آن بیاموزد؛ هم‌چنین اجازه می‌دهد تا یک مجموعه داده را در یادگیری ماشینی برچسب‌گذاری کنید و در یادگیری نظارت شده، برچسب‌گذاری مجموعه داده، بخش مهمی از پیش‌پردازش داده است؛ بنابراین برای طبقه‌بندی می‌تواند ورودی و خروجی را برچسب‌گذاری کند. برچسب‌گذاری داده‌ها فرآیند مهمی است؛ زیرا می‌تواند قبل از استفاده از آن در مدل آموزشی، زمینه و مفهوم را به داده‌ها اضافه کند، به طوری که برچسب‌گذاری داده‌ها به ما کمک می‌کند تا زمانی که می‌خواهیم عامل مقیاس‌پذیری و فاکتور کیفیت را بهبود بخشیم، رویکرد صحیحی را انتخاب کنیم. برچسب‌گذاری داده‌ها فرآیند شناسایی داده‌های خام و برچسب‌گذاری آن است، به همین منظور در این پژوهش با استفاده از نظرخواهی از چندین نفر برچسب گذاری نظرات به دو صورت مثبت و منفی(۱ و ۰) انجام شده است. این مجموعه داده دارای ۱۰۲۴ نمونه از نظرات کاربران سایت سهامیاب می باشد که هرکدام از این برچسب ها قطبیت نظرات در مورد شرکت های فعال بورس را بیان می کند.

۳-۳- پیش پردازش داده

باتوجه به اینکه داده‌هایی که در این مقاله استفاده شده است از بستر اینترنت جمع آوری شده‌اند، ساختار مناسبی برای ورود به شبکه‌های عصبی عمیق ندارند، بنابراین قبل از ورود به سیستم باید به شکلی قابل فهم برای الگوریتم‌ها تبدیل شوند. در همین راستا تغییراتی درقالب پیش پردازش بر روی این داده‌ها اعمال می‌شود. از آن جایی‌که داده‌های این پژوهش از سایت سهامیاب گردآوری شده و حاصل تعامل کاربران می باشد در نتیجه ساختار مناسبی برای فرآیند تحلیل احساسات و ورود به شبکه عصبی ندارد، بنابراین انجام عملیات پیش‌پردازش بر روی این داده ها ضروری است .

پیش پردازش داده (Data Preprocessing) به مراحلی گفته می‌شود که در آن داده‌ها برای داده کاوی آماده می‌شود. باتوجه به اینکه داده های تولید شده توسط کاربران فضای اینترنت، دارای نواقصی هستند که باعث به وجود آمدن مشکلاتی در داده کاوی می‌شود؛ بنابراین قبل از هر گونه پردازش روی این داده‌ها، می‌بایست عملیات پیش پردازش (حPreprocessing) صورت گیرد که طی آن داده‌های خام با اعمال تغییراتی مانند تبدیل کردن اعداد به کلمات و یا حذف کردن اعداد از داده‌های متنی، پاک کردن علائم و فاصله خالی و... به داده های مناسب به منظور فرآیند تحلیل احساسات تبدیل می‌شوند.

یکی از کتابخانه‌های اصلی پایتون برای آماده‌سازی و پیش‌پردازش داده‌ها، پانداس (Pandas) است که کارایی بالا، ساختاری با قابلیت استفاده آسان و ابزارهای تحلیل داده برای زبان برنامه‌نویسی پایتون را فراهم می‌کند. توابع گوناگون Pandas به ساده‌سازی فرایند پیش پردازش داده‌ها کمک قابل توجهی می‌کنند. در واقع، می‌توان گفت پانداس یک کتابخانه قدرتمند برای تحلیل، و پیش‌پردازش (PreProcessing) داده‌ها است. این کتابخانه می‌تواند داده‌ها را با بهره‌گیری از ساختارهای Series و DataFrame که ارائه می‌کند، به قالبی که برای تحلیل داده‌ها مناسب هستند، مبدل سازد.

یک Series مشابه با آرایه یک‌بُعدی است. Series می‌تواند داده‌ها از هر نوعی را ذخیره کند. مقادیری که در Series قرار می‌گیرند قابل تغییر هستند؛ اما اندازه Series پانداس، غیر قابل تغییر است. به اولین عنصر در Series، اندیس صفر تخصیص داده می‌شود و اندیس آخرین عنصر در Series برابر با N-1 است که در آن، N تعداد کل عنصرهای موجود در سری است. ساختار داده دیتافریم (DataFrame) در پانداس را می‌توان به عنوان یک جدول در نظر گرفت. دیتافریم، داده‌ها را در سطرها و ستون‌ها سازمان‌دهی می‌کند و از آن‌ها یک ساختار داده دو بُعدی می‌سازد. ستون‌ها می‌توانند حاوی مقادیری از انواع گوناگون باشند و در عین حال، اندازه دیتافریم قابل تغییر است؛ بنابراین می‌توان آن را ویرایش کرد. برای ساخت دیتافریم، می‌توان کار را از پایه شروع کرد و یا ساختار داده‌هایی مانند آرایه‌های (Numpy) را به یک دیتافریم تبدیل کرد. ساختاری که در اینجا استفاده شده است ساختار DataFrame است.

پس از این‌که فایل اکسل مجموعه داده ایجاد شد برای انجام عملیات پیش پردازش به برنامه وارد می شود. این فایل اکسل بیانگر نظرات فارسی، نظرات انگلیسی و برچسب متناظر با هرکدام از این نظرات است. ابتدا با استفاده از کتابخانه Pandas فایل مجموعه داده خوانده و آماده پیش‌پردازش می‌شود. به‌این صورت که تعدادی لیست واژه (Stopword) ایجاد شده حذف می شوند و با استفاده از تابع Replace علائم نگارشی موجود در متن نظرات پاک می شوند. سپس یک لغت‌نامه از کلمات موجود ساخته می‌شود و بعد از این‌که عملیات نرمال‌سازی روی این کلمات صورت گرفت آماده ورود به الگوریتم های یادگیری می‌شوند.

از آن‌جایی‌که ورودی شبکه‌های عصبی به صورت عدد است، لغت نامه‌ای که در این پژوهش استفاده می شود با استفاده از اندیس متناظر کلمات ایجاد می شود. به این منظور برای هرکدام از لغات یک اندیس عددی در نظر گرفته می شود. هرنظر با لیستی از اعداد صحیح بیان می‌شود. باتوجه به این‌که طول داده‌های ورودی شبکه‌های عصبی باید یکسان باشد با استفاده از تابعPad-Sequence از کتابخانه Keras عملیات نرمال‌سازی روی نظرات انجام می شود و طول تمام نظرات یکسان و برابر عدد ۳۰ می‌شود.

مرحله آخر تقسیم داده‌ها به سه دسته آموزش، اعتبارسنجی و آزمون است. دراین پژوهش داده‌ها به نسبت‌های ۶۴ درصد ،۱۶ درصد و ۲۰ درصد برای هرکدام از دسته‌های آموزش، اعتبارسنجی و آزمون تقسیم شده است. به این صورت که با استفاده از تابع Train_Test_Split از کتابخانه sklearn، ۲۰ درصد داده‌ها برای آزمون (Test) و ۸۰ درصد باقیمانده با نسبت ۸۰ به ۲۰ برای آموزش (Train) و اعتبارسنجی (Validation) تقسیم بندی شده‌اند.

۴- الگوریتم‌های سنتی یادگیری ماشین

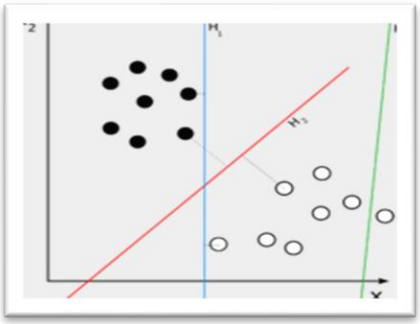
یکی از اهداف این پژوهش مقایسه مدل پیشنهادی خود با مدل‌های سنتی یادگیری ماشین جهت بررسی دقت و کارایی است. برای بررسی این هدف تعدادی از الگوریتم‌های سنتی یادگیری ماشین مانند KNN،SVM، درخت تصمیم و بیزین ساده با استفاده از کتابخانه Sklearn پیاده سازی می‌شوند. Scikit Learn از کتابخانه‌های متن‌باز، مفید، پرکاربرد و قدرتمند در زبان برنامه‌نویسی پایتون است که برای اهداف یادگیری ماشین به کار می‌رود. این کتابخانه ابزارهای کاربردی زیادی به منظور یادگیری ماشین و مدل‌سازی آماری داده‌ها هم‌چون طبقه‌بندی (Classification)، رگرسیون، خوشه‌بندی و کاهش ابعاد فراهم می‌کند. این کتابخانه که به طور عمده توسط زبان پایتون ارائه شده، بر پایه‌ی کتابخانه‌های Scipy،Numpy و Matplotlib طراحی شده است.

۴-۱- ماشین بردار پشتیبان

ماشین بردار (Support Vector Machin) یکی از روش‌های یادگیری با نظارت است که از آن برای طبقه‌بندی و رگرسیون استفاده می‌کنند. این روش از روش‌های نسبتا جدیدی است که در سال‌های اخیر کارایی خوبی نسبت به روش‌های قدیمی‌تر برای طبقه‌بندی نشان داده است. پیش از آن که در صحبت و بررسی در مورد ماشین‌های بردار پشتیبان فراتر برویم؛ باید با یک سری از مفاهیم مربوطه آشنایی مختصری پیدا کنیم

۴-۱-۱- خط یا ابر صفحه جداکننده

هدف پیدا کردن بهترین خط (ابر صفحه) است که دو دسته را از هم جدا کند. مطابق شکل۱ H3 (سبز) دو دسته را از هم جدا نمی‌کند اما H1 (آبی) دو دوسته را از هم با حاشیه‌ای کوچک جدا می‌کند و H2 (قرمز) دو دسته را با حداکثر حاشیه از هم جدا می‌سازد.



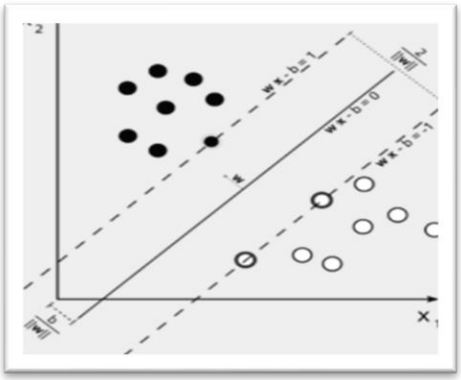
شکل ۱- خط یا ابرصفحه جداکننده

۴-۱-۲- حداکثر حاشیه

بر طبق قضیه‌ای در تئوری یادگیری اگر مثاله‌ای آموزشی به درستی دسته‌بندی شده باشند، از بین جداسازهای خطی، آن جداسازی که حاشیه داده‌های آموزشی را حداکثر می‌کند، خطای تعمیم را حداقل خواهد کرد. این روش به نظر مطمئن‌ترین راه است و به طور تجربی به خوبی جواب داده و البته تئوری‌هایی برمبنای VC dimension وجود دارد که مفید بودن آن را اثبات می‌کند. دلیل این‌که SVM روی بزرگترین مرز برای Hyperplane پافشاری می‌کند این است که قضیه قابلیت عمومیت بخشیدن به الگوریتم را بهتر تامین می‌کند. این نه تنها به کارایی طبقه‌بندی و دقت آن روی داده‌های آزمایشی کمک می‌کند، فضا را نیز برای طبقه‌بندی بهتر داده‌های آتی مهیا می‌کند. به طور حسی آن مرزی که به صورت بخشی از فضا تعریف می‌شود یا همان تفکیک بین دو کلاس به وسیله Hyperplane تعریف می‌شود. همین تعریف هندسی به ما اجازه می‌دهد تا کشف کنیم که چگونه مرزها را بیشینه کنیم ولو این‌که تعداد بیشماری Hyperplane داشته باشیم و فقط تعداد کمی، شایستگی راه حل برای SVM دارند.

۴-۱-۳- بردار پشتیبان

شکل ۲ ابرصفحه‌ای با حداکثر حاشیه برای یک ماشین بردار پشتیبان که با نمونه داده‌هایی از دو دسته یادگرفته شده است را نمایش می دهد. داده‌هایی که بر روی آبرصفحه حاشیه قرار دارند بردارهای پشتیبان نام دارند به عبارتی نزدیک‌ترین داده‌های آموزشی به ابر صفحه‌های جداکننده بردار پشتیبان نامیده می‌شوند.



شکل۲- بردار پشتیبان

۴-۱-۴- ماشین بردار پشتیبان خطی

ماشین بردار پشتیبان یک روش یادگیری نسبتاً جدید است که اغلب برای کلاس‌بندی باینری مورد استفاده واقع می‌شود. فرض کنید L مشاهده داریم که هر مشاهده مشتمل بر زوج‌هایی است که در آن بردار ورودی و یک مقدار دو وضعیتی (۱- یا ۱+) وجود دارد. ایده ماشین بردار پشتیبان می‌کوشد، ابر صفحاتی در فضا رسم کند که عمل تمایز نمونه‌های کلاس‌های مختلف داده‌ها را به طور بهینه انجام دهد.

مبنای کاری دسته بندی‌کننده SVM دسته‌بندی خطی داده‌ها است و در تقسیم خطی داده‌ها سعی می‌کنیم خطی را انتخاب کنیم که حاشیه اطمینان بیش‌تری داشته باشد. حل معادله پیدا کردن خط بهینه برای داده‌ها به وسیله روش‌های QP که روش‌های شناخته شده‌ای در حل مسائل محدودیت دار هستند صورت می‌گیرد. قبل از تقسیم خطی برای این‌که ماشین بتواند داده‌های با پیچیدگی بالا را دسته‌بندی کند داده‌ها را با استفاده از تابع Phi به فضای با ابعاد خیلی بالاتر می‌بریم. برای این‌که بتوانیم مسئله ابعاد خیلی بالا را با استفاده از این روش‌ها حل کنیم از قضیه دوگانی لاگرانژ برای تبدیل مسئله مینیمم‌سازی مورد نظر به فرم دوگانی آن که در آن به جای تابع پیچیده Phi که ما را به فضایی با ابعاد بالا می‌برد، تابع ساده‌تری به نام تابع هسته که ضرب برداری تابع Phi است ظاهر می‌شود، استفاده می‌کنیم. از توابع هسته مختلفی از جمله هسته‌های نمایی، چند جمله‌ای و سیگموید می‌توان استفاده نمود. با فرض این‌که دسته‌ها به صورت خطی جداپذیر باشند، ابرصفحه‌هایی با حداکثر حاشیه (Maximum Margin) را به دست می‌آورد که دسته‌ها را جدا کنند. در مسایلی که داده‌ها به صورت خطی جداپذیر نباشند، داده‌ها به فضای با ابعاد بیشتر نگاشت پیدا می‌کنند تا بتوان آن‌ها را در این فضای جدید به صورت خطی جدا نمود.

در یک فرآیند یادگیری که شامل دو کلاس می‌باشد، هدف SVM پیدا کردن بهترین تابع برای طبقه‌بندی می‌باشد به نحوی که بتوان اعضای دو کلاس را در مجموعه داده‌ها از هم تشخیص داد. معیار بهترین طبقه‌بندی به صورت هندسی مشخص می‌شود، برای مجموعه داده‌هایی که به صورت خطی قابل تجزیه هستند.

ماشین بردار پشتیبان از جمله متدهای قدرتمند طبقه‌بندی نظارت شده است. در ابعاد بالا انعطاف‌پذیر است و در مسائل طبقه‌بندی به طور گسترده استفاده می‌شود و از نظر مصرف حافظه کاراست؛ این الگوریتم با ایجاد بردارهایی درفضای داده ها به بهترین شکل ممکن تفکیک داده ها را انجام می دهد. ماشین بردار پشتیبان در ابتدا ابرصفحاتی را تولید می‌کند که نقاط را به شکل صحیح تقسیم می‌کند. سپس میان ابرصفحات آن ابرصفحه‌ای را بر می‌گزیند که نقاط را به بهترین شکل ممکن جدا می‌کند.

دو پارامتر به کار رفته در این الگوریتم SVC از کلاس SVM و Kernel از جنس String هستند. طبقه‌بندی از طریق کلاس SVC صورت می گیرد. SVC طبقه‌بندی چند کلاسه را از طریق مقایسه و طبقه‌بندی به صورت یک به یک (یک برچسب در مقایسه با سایر نمونه‌ها) انجام می‌دهد. پارامتر kernel مقداری اختیاری است و پیش فرض آن rbf است. این پارامتر تعیین کننده‌ی نوع kernel به کار گرفته شده در الگوریتم است و می‌تواند یکی از مقادیر linear، poly، rbf، sigmoid و precomputed را بگیرد که مقدار پیش‌فرض rbf است.

همانطور که شکل ۳ نشان می دهد صحت اجرای این الگوریتم بر روی داده‌های پژوهش برابر با ۶۱ درصد است.

	precision	recall	f1-score	support
Negative	0.50	0.22	0.30	79
Positive	0.64	0.87	0.73	126
accuracy			0.61	205
macro avg	0.57	0.54	0.52	205
weighted avg	0.58	0.61	0.57	205

شکل ۳- نمایی از خروجی الگوریتم SVM

الگوریتمی که برای دسته‌بندی داده‌ها استفاده شود، در نهایت هر نمونه عضو یکی از این دو دسته (Class) دسته‌بندی خواهد شد؛ بنابراین برای هر نمونه داده، یکی از چهار حالتی که در ادامه بیان شده، ممکن است اتفاق بیفتد.

- نمونه عضو دسته مثبت باشد و عضو همین کلاس تشخیص داده شود. مثبت صحیح یا (True Positive)
- نمونه عضو کلاس مثبت باشد و عضو کلاس منفی تشخیص داده شود. منفی کاذب یا (False Negative)
- نمونه عضو کلاس منفی باشد و عضو همین کلاس تشخیص داده شود. منفی صحیح یا (True Negative)
- نمونه عضو کلاس منفی باشد و عضو کلاس مثبت تشخیص داده شود. مثبت کاذب یا (False Positive)

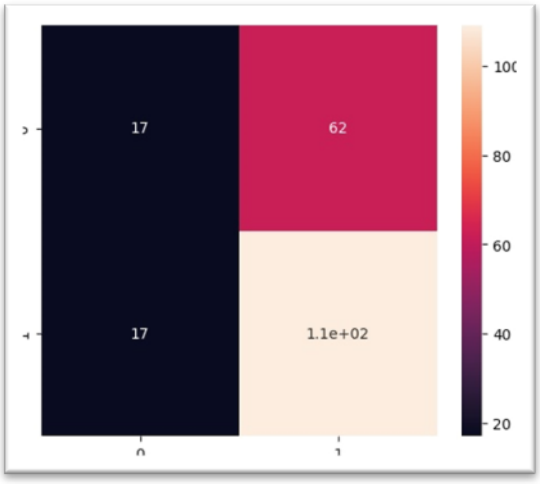
پس از اجرای الگوریتم دسته‌بندی، با توجه به توضیحات و تعاریف ذکر شده، می‌توان عملکرد یک طبقه‌بند را به کمک جدولی به شکل جدول ۲ با عنوان جدول یا ماتریس درهم ریختگی (Confusion Matrix) بررسی کرد. ماتریس درهم ریختگی، نتایج حاصل از طبقه‌بندی را بر اساس اطلاعات واقعی موجود، نمایش می‌دهد. حال بر اساس این مقادیر می‌توان معیارهای مختلف ارزیابی دسته بند و اندازه‌گیری دقت را تعریف کرد. پارامتر صحت (Accuracy)، متداول‌ترین، اساسی‌ترین و ساده‌ترین معیار اندازه‌گیری کیفیت یک دسته‌بند است و عبارت است از میزان تشخیص صحیح دسته‌بند در مجموع دو دسته. این پارامتر در واقع نشان‌گر میزان الگوهایی است که درست تشخیص داده شده‌اند.

جدول ۲- ماتریس درهم ریختگی

		برچسب پیش بینی شده	
		منفی	مثبت
برچسب شناخته شده	منفی	TN	**FN**
	مثبت	FP	**TP**

در این پژوهش ماتریس درهم ریختگی با نمودارHeat Map (نقشه حرارتی) نمایش داده می‌شود. Heat Map یک نمایش گرافیکی داده است. نقشه های حرارتی برای شناسایی الگوها در مقادیر زیاد داده در یک نگاه بسیار مفید هستند. نقشه‌های حرارتی اغلب نقطه شروع خوبی برای تحلیل های پیچیده‌تر است؛ اما همچنین یک تکنیک تجسم چشم نواز است که آن را به ابزاری مفید برای ارتباط تبدیل می‌کند.

این نقشه اساساً شبکه‌ای از مربع‌های رنگی است که هر مربع یا صندوقچه، تقاطع مقادیر دو متغیر را نشان می‌دهد که در امتداد محورهای افقی و عمودی امتداد دارند. محور عمودی مقادیر واقعی و محور افقی مقادیر پیش بینی شده توسط مدل را نمایش می‌دهد. ماهیت دوبعدی نقشه رنگی به خاطر به نمایش گذاشتن اطلاعات ماتریس می‌باشد. شکل ۴ نمودار Heat Map الگوریتم SVM را نشان می‌دهد.



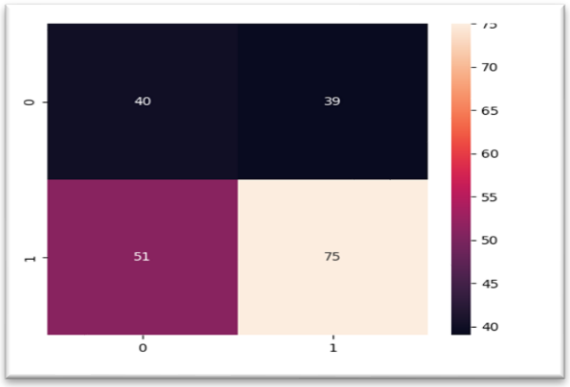
شکل ۴- نمودار Heat Map لگوریتم SVM

۴-۲- الگوریتم درخت تصمیم

الگوریتم درخت تصمیم از محبوب‌ترین و کاربردی‌ترین الگوریتم‌ها در زمینه طبقه‌بندی داده‌ها است. این الگوریتم با ساختار درختی خود به جداسازی و طبقه بندی داده‌ها می‌پردازد. در این مقاله ابتدا یک نمونه از کلاس این الگوریتم ساخته می‌شود سپس فرآیند اجرا و آزمون انجام می‌شود. همانطور که شکل ۵ نشان می‌دهد صحت اجرای این الگوریتم بر روی داده‌های پژوهش برابر با ۵۶ درصد است و شکل ۶ نمودار Heat Map الگوریتم درخت تصمیم را نشان می‌دهد.

	precision	recall	f1-score	support
Negative	0.44	0.51	0.47	79
Positive	0.66	0.60	0.62	126
accuracy			0.56	205
macro avg	0.55	0.55	0.55	205
weighted avg	0.57	0.56	0.57	205

شکل ۵- نمایی از خروجی الگوریتم درخت تصمیم



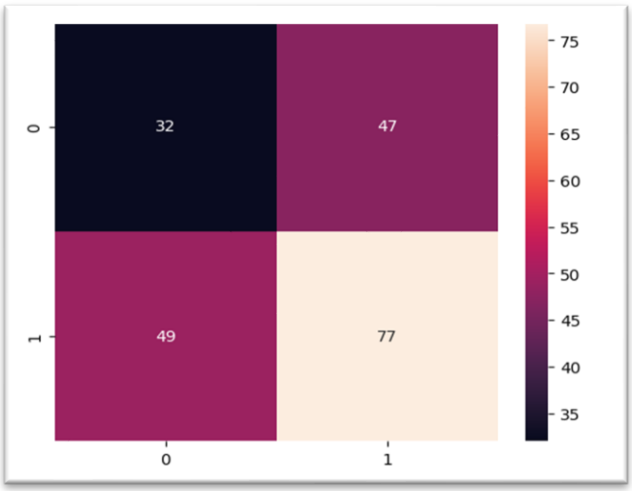
شکل ۶- نمودار Heat Map الگوریتم درخت تصمیم

۴-۳- الگوریتم بیزین ساده

الگوریتم بیزین یکی از الگوریتم‌های سنتی یادگیری ماشین در فرآیند طبقه‌بندی داده است که براساس تکنیک‌های آماری به طبقه بندی این داده‌ها می‌پردازد. در این مقاله به دلیل گسسته بودن داده‌ها از مدل MultinomialNB برای این الگوریتم استفاده شده است. همانطورکه شکل ۷ نشان می‌دهد صحت بدست آمده از اجرای این الگوریتم بر روی داده‌ها ۵۳ درصد است. شکل ۸ نمودار Heat Map الگوریتم بیزین ساده را نمایش می‌دهد.

	precision	recall	f1-score	support
Negative	0.40	0.41	0.40	79
Positive	0.62	0.61	0.62	126
accuracy			0.53	205
macro avg	0.51	0.51	0.51	205
weighted avg	0.53	0.53	0.53	205

شکل ۷- خروجی الگوریتم بیزین ساده



شکل ۸- نمودار Heat Map الگوریتم بیزین ساده

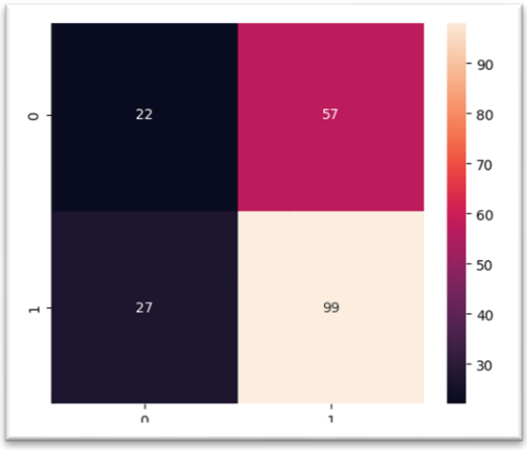
۴-۴- الگوریتم نزدیک‌ترین همسایه (KNN)

برای پیاده‌سازی الگوریتم KNN در پایتون از کتابخانه Scikit-Learn استفاده می‌شود که یکی از پرکاربردترین الگوریتم‌ها درحوزه یادگیری ماشین سنتی است که هم برای مسایل رگرسیون و هم طبقه‌بندی کاربرد دارد. همانطور که شکل ۹ نشان می‌دهد صحت اجرای این الگوریتم بر روی داده‌های این مقاله ۵۹ درصد است.

	precision	recall	f1-score	support
Negative	0.45	0.28	0.34	79
Positive	0.63	0.79	0.70	126
accuracy			0.59	205
macro avg	0.54	0.53	0.52	205
weighted avg	0.56	0.59	0.56	205

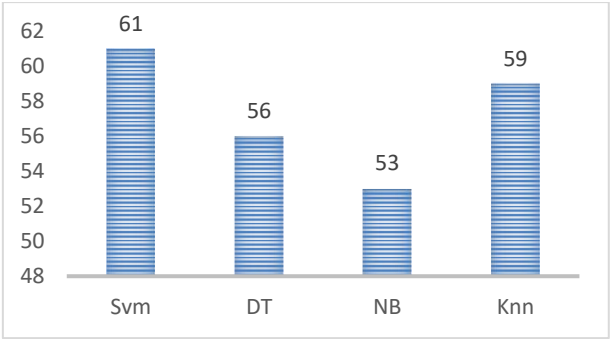
شکل ۹- خروجی الگوریتم KNN

شکل ۱۰ نمودار Heat Map الگوریتم KNN را نشان می‌دهد.



شکل ۱۰- نمودار Heat Map الگوریتم KNN

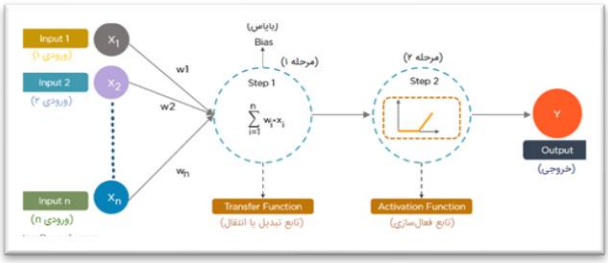
شکل ۱۱ دقت الگوریتم‌های یادگیری ماشین سنتی را مقایسه می‌کند و نشان می‌دهد الگوریتم SVM بیشترین دقت را داشته است.



شکل ۱۱- مقایسه دقت الگوریتم‌های یادگیری ماشین سنتی

۵- مدل‌های شبکه عصبی

شبکه‌های عصبی (Neural Network) در واقع دسته‌ای از الگوریتم‌های یادگیری ماشین هستند که برای شناسایی و تشخیص الگوها به کار می‌روند. یک شبکه عصبی توسط ورودی‌ها آموزش داده می‌شود و شامل لایه ورودی، پنهان و خروجی است و هر کدام از عصب ها دارای مقدار آستانه و تابع فعال‌سازی می‌باشند که منجر به خروجی می‌شوند. نتیجه به دست آمده با خروجی مورد انتظار مقایسه می‌شود که این دو مقدار باید نزدیک به هم باشند. مدل یاد می‌گیرد که وزن‌ها و مقدار آستانه را طوری تنظیم کند که خروجی درست دریافت کند. یک شبکه عصبی، مانند مغز انسان ساختار یافته است و از نورون‌های مصنوعی تشکیل شده است که با عنوان "گره" نیز شناخته می‌شود. این گره‌ها در سه لایه ورودی، مخفی و خروجی در کنار هم چیده شده‌اند. شکل ۱۲ ساختار شبکه عصبی را نمایش می‌دهد.



شکل ۱۲- ساختار شبکه عصبی

داده‌ها، اطلاعاتی را در قالب ورودی به هر گره ارائه می‌دهند. گره، ورودی‌ها را در وزن‌های تصادفی ضرب می‌کند، آن‌ها را محاسبه کرده و یک بایاس به آن اضافه می‌کند. در نهایت، توابع غیرخطی، که به آن‌ها "توابع فعال سازی" گفته می شود، برای تعیین این‌که کدام نورون شلیک کند، اعمال می‌شوند.

دراین مقاله الگوریتم های شبکه عصبی با استفاده از کتابخانه Keras ساخته می‌شوند که یک کتابخانه یادگیری عمیق منبع باز و یکی از بهترین کتابخانه‌های پایتون محسوب می‌شود. این کتابخانه آزمایش سریع با مدل‌های یادگیری عمیق را امکان‌پذیر می‌کند و از شبکه‌های کانولوشنال و شبکه‌های تکراری و همچنین ترکیبی از این دو پشتیبانی می کند. این کتابخانه پایتون به طور گسترده برای برنامه‌هایی مانند تشخیص و پردازش تصویر، پردازش زبان طبیعی و تشخیص گفتار استفاده می‌شود.

یکی دیگر از موارد استفاده بالقوه برای Keras برای پردازش زبان طبیعی (NLP) است. از Keras می‌توان برای ساخت مدل‌هایی استفاده کرد که می‌توانند متن زبان طبیعی را درک و تفسیر کنند. این مدل‌ها را می‌توان با مجموعه‌ای از متن، مانند مقاله‌های خبری یا کتاب، آموزش داد و سپس برای تحلیل و تفسیر متن جدید استفاده کرد. این می‌تواند توسط یک سیستم خودکار برای پردازش و درک پرس‌وجوها یا دستورات زبان طبیعی استفاده شود.

Keras یک API سطح بالا برای ساخت مدل های یادگیری عمیق است. مجموعه ای از کلاس ها و عملکردها را برای کمک به ایجاد و آموزش مدل های یادگیری عمیق ارائه می‌دهد.

کلاس و توابع اصلی در کتابخانه Keras عبارت‌اند از:

- Model• این کلاس برای ایجاد مدل شبکه عصبی که از لایه‌ها تشکیل شده است استفاده می‌شود.
- Sequential• این کلاس برای ایجاد یک پشته خطی از لایه‌ها استفاده می‌شود.
- Dense• از این کلاس برای ایجاد یک لایه متصل متراکم استفاده می‌شود.
- Activation• این کلاس برای افزودن یک تابع فعال سازی به یک لایه استفاده می‌شود.
- Convolutional• از این کلاس برای ایجاد یک لایه کانولوشنال استفاده می‌شود.
- MaxPooling• این کلاس برای ایجاد یک لایه Max pooling استفاده می‌شود.
- Dropout• این کلاس برای ایجاد لایه Dropout استفاده می‌شود.
- Flatten• از این کلاس برای صاف کردن لایه ورودی استفاده می‌شود.
- Optimizers• این کلاس برای تعریف بهینه‌ساز مورد استفاده برای آموزش مدل استفاده می‌شود.

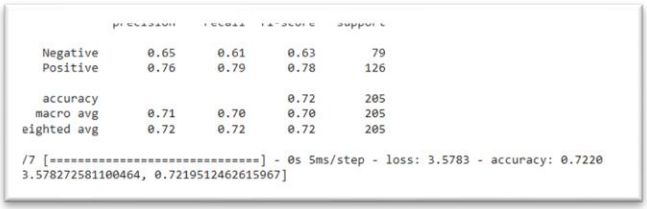
•Losses این کلاس برای تعریف تابع ضرر مورد استفاده برای محاسبه خطای مدل استفاده می‌شود.

•Metrics از این کلاس برای تعریف معیارهای لازم استفاده می‌شود.

دراین قسمت مدل‌های شبکه عصبی LSTM وBERT که دراین مقاله استفاده شده اند شرح داده می‌شوند. نظر به این‌که درتمام این مدل ها لایه تعبیه ساز یکسان است، پیاده‌سازی این لایه نیز توضیح داده می شود. این لایه در اولین لایه معماری مدل‌های شبکه عصبی عمیق قرار گرفته و بعنوان لایه ورودی درنظر گرفته می‌شود. برای پیاده سازی لایه تعبیه ساز در پایتون از لایه Embedding که در کتابخانه Keras تعریف شده است، استفاده می‌شود. پارامترهای این لایه که شامل تعداد لغات، اندازه بردار متراکم و طول نظرات که همان داده‌های ورودی هستند مقداردهی می‌شوند. مقدار ۳۰ برای اندازه بردار متراکم و طول نظرات درنظر گرفته شده است.

۵-۱- مدل LSTM

الگوریتم LSTM یک نوع شبکه عصبی بازگشتی است که می‌تواند از اطلاعات گذشته در پیش‌بینی آینده استفاده کند. این شبکه دارای یک حافظه داخلی است که می‌تواند اطلاعات مربوطه را نگه دارد و اطلاعات نامربوط را فراموش کند. پیاده‌سازی الگوریتم LSTM در پایتون نیاز به استفاده از کتابخانه‌هایی مانند تنسورفلو (Tensorflow) یا پایتورچ (PyTorch) دارد که امکان ساخت و آموزش شبکه‌های عصبی را فراهم می‌کنند. همچنین باید معماری و پارامترهای شبکه LSTM را با توجه به مسئله مورد نظر تعیین کرد. برای این کار، می‌توان از الگوریتم‌های فراابتکاری (Metaheuristic) مانند الگوریتم ژنتیک (Genetic Algorithm) که می‌توانند به بهینه‌سازی پارامترهای شبکه LSTM کمک کنند استفاده کرد. این مدل به صورت ترتیبی ساخته می‌شود. لایه تعبیه ساز به اولین لایه اضافه شده و سپس یک لایه LSTM به تعداد ۱۶ واحد می سازد و بعد از آن یک لایه ی Dense با یک نورون برای تولید خروجی قرار می‌دهد. تابع Sigmoid تابع فعال ساز در خروجی این مدل است. در مرحله آخر مدل با بهینه ساز adam ، تابع ضرر binary-crossentropy و معیار Accuracy کامپایل می‌شود. همانطور که شکل ۱۳ نشان می‌دهد صحت عملکرد این مدل برای تشخیص قطبیت نظرات کاربران سایت سهامیاب ۷۲ درصد است.



	precision	recall	f1-score	support
Negative	0.65	0.61	0.63	79
Positive	0.76	0.79	0.78	126
accuracy			0.72	205
macro avg	0.71	0.70	0.70	205
eighted avg	0.72	0.72	0.72	205

/7 [==============================] - 0s 5ms/step - loss: 3.5783 - accuracy: 0.7220
3.578272581100464, 0.7219512462615967]

شکل ۱۳- نمایی از خروجی الگوریتم LSTM

۵-۲- بهینه‌سازی الگوریتم LSTM با ژنتیک

الگوریتم ژنتیک یک تکنیک بهینه‌سازی است که الهام‌گرفته از فرایند تکامل در طبیعت است. این الگوریتم با تولید و انتخاب جمعیت‌هایی از جواب‌های محتمل برای یک مسئله، سعی می‌کند به جواب بهینه برسد. پیاده‌سازی الگوریتم ژنتیک بر LSTM به این معنی است که می‌خواهیم معماری و پارامترهای شبکه LSTM را با استفاده از الگوریتم ژنتیک تنظیم کنیم. به این ترتیب، می‌توانیم بهترین شبکه LSTM را برای پیش‌بینی سری‌های زمانی چند متغیره پیدا کنیم. برای این کار، باید ابتدا جمعیت اولیه‌ای از شبکه‌های LSTM با معماری‌ها و پارامترهای مختلف تولید کنیم. سپس، باید تابع هدف و تابع برازندگی را برای ارزیابی عملکرد شبکه‌ها تعریف کنیم. در نهایت، باید عملگرهای ژنتیک مانند جهش، جابجایی و ترکیب را بر روی جمعیت اعمال کنیم تا نسل‌های جدیدی از شبکه‌های LSTM را تولید کنیم.

به طور خلاصه، می‌توان گفت که الگوریتم ژنتیک بر LSTM یک روش بهینه‌سازی است که با شبیه‌سازی فرایند تکامل، می‌تواند بهترین شبکه LSTM را برای پیش‌بینی سری‌های زمانی چند متغیره انتخاب کند. این روش می‌تواند به شبکه LSTM کمک کند تا از اطلاعات گذشته بهتر استفاده کند و پیش‌بینی‌های دقیق‌تری ارائه دهد. انتخاب هایپر پارامتر مناسب یکی از مهم ترین مسائل درپیاده سازی مدل LSTM در مرحله ی آموزش است، دست یافتن به این هایپرپارامتر که منجر به یک الگوریتم بهینه شود کاری پیچیده است. در این مقاله پس از پیاده سازی مدل LSTM ، به منظور بهینه سازی عملکرد این مدل از الگوریتم فرا ابتکاری ژنتیک استفاده می‌شود. بدین منظور ابتدا مقادیر مجاز برای هر پارامتر تعیین می‌شود. مقادیر مجاز برای این الگوریتم به شرح زیر است:

•Unit[۶۴ ,۳۲ ,۱۶ ,۸] :

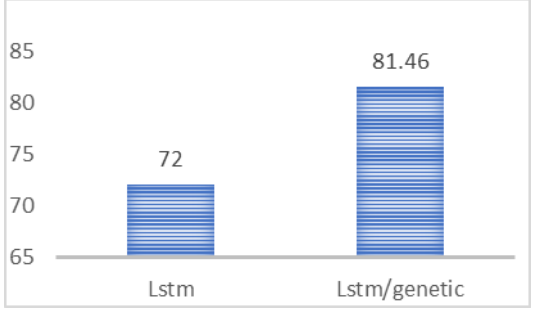
•Activation: [“sigmoid”, “relu”, “selu”, “elu”,” softmax”, “softplus”, “tanh”]

•Optimizer: [“adamax”, “adam”, “rmspropo”, “nadam”]

•Loss: [“binary_crossentropy”,” mse”, “mae”]

•Batch[۶۴ ,۳۲ ,۱۶ ,۸ ,۴ ,۲] :

با استفاده از الگوریتم فرا ابتکاری ژنتیک هایپر پارامترهای تعداد واحد، تابع فعالساز، بهینه ساز، تابع ضرر و اندازه بسته ها بهینه می‌گردد. سپس با استفاده از این مقادیر مجاز یک جمعیت اولیه به تعداد ۱۰ کروموزوم ساخته می‌شود، هرکدام از ژن‌های یک کروموزوم شامل یک مقدار مجاز از هایپر پارامترهاست که انتخاب این مقادیر به صورت تصادفی است؛ بنابراین هرکدام از کروموزوم‌ها شامل ترکیبی بهینه از هایپر پارامتر می باشد که در آموزش مدل استفاده می‌شود. سپس ارزیابی کروموزوم ها با تابع برازش انجام می‌شود به این صورت که مدل LSTM به ازای هرکروموزوم یک بار اجرا می‌شود و بعد از آموزش با داده‌های آزمون ارزیابی می‌شود. بعد از ارزیابی جمعیت با استفاده از چرخ رولت کروموزوم های ارزنده بعنوان والد انتخاب می‌شوند. در مرحله بعدی کروموزوم‌های والد با روش چند نقطه ای باهم ترکیب شده و کروموزوم جدید که هایپر پارامترهای والدین را به ارث برده ایجاد می‌شود و در نهایت این کروموزوم بعد از جهش به جمعیت اضافه می‌شود. سپس شرط پایانی بررسی می شود به این صورت که تازمانی که صحت اجرای مدل به مقدار آستانه که در این جا ۹۹ درصد است نرسیده باشد و یا تعداد نسل‌های الگوریتم که در این مقاله به تعداد ۵۰ نسل است به پایان نرسیده باشد، اجرای الگوریتم ادامه می‌یابد در غیراین صورت الگوریتم پایان می‌یابد و ژن های موجود در آن کروموزوم به عنوان هایپر پارامتر بهینه معرفی شده و در مرحله بعد ارزیابی می‌شوند و بهترین عضو آن‌ها به عنوان هایپر پارامتر ارزنده معرفی می‌شود و مدل LSTM را بهینه می‌کند. شکل ۱۴ نشان می‌دهد که صحت عملکرد این الگوریتم بر روی داده‌های این مقاله ۸۱/۴۶ درصد است در حالیکه پس از اجرای الگوریتم LSTM به تنهایی صحت ۷۲ درصد بدست آمد.



شکل ۱۴- نمودار مقایسه LSTM و پیاده سازی LSTM با الگوریتم ژنتیک

جدول ۳ بهترین عملکرد به همراه هایپر پارامترهای بهینه آن را نشان می‌دهد.

جدول ۳- بهترین عملکرد ومقادیر هایپر پارامترهای بهینه

صحت	نسل	هایپرپارامترهای بهینه					
%	۲۲	Batch- size	loss	optimizer	activation	Unit2	Unit1
۸۱/۴۶		۴	binary_crossentropy	rmsprop	sigmoid	۳۲	۸

۳-۵- مدل BERT

پیاده‌سازی الگوریتم BERT با زبان پایتون یکی از راه‌هایی است که می توان از این الگوریتم پیشرفته برای پردازش زبان طبیعی استفاده کرد. یکی از روش‌هایی است که می‌توانیم از این الگوریتم پیشرفته برای درک بهتر معنای کلمات و عبارات جستجو شده توسط کاربران استفاده کنیم. این الگوریتم از یک مدل ترنسفورمر دو طرفه استفاده می‌کند که می‌تواند با توجه به موقعیت و ارتباط کلمات در یک جمله، معنای آن‌ها را تشخیص دهد.

برای این کار نیاز به دانستن مفاهیم اساسی پایتون، تنسورفلو و ترنسفورمرز است. تنسورفلو یک کتابخانه یادگیری عمیق است که این امکان را فراهم می‌کند تا مدل‌های پیچیده را با استفاده از گراف‌های محاسباتی بسازیم. ترنسفورمرز یک کتابخانه است که شامل مجموعه‌ای از مدل‌های پردازش زبان طبیعی مبتنی بر BERT و مدل‌های مشابه است. با استفاده از این کتابخانه، می‌توانیم BERT را به راحتی بارگذاری کنیم، آموزش دهیم و ارزیابی کنیم.

کتابخانه تنسورفلو در پایتون؛ یک کتابخانه ریاضی نمادین است و با استفاده از داده ها و برنامه نویسی در فعالیت های مختلف شبکه‌های عصبی استفاده می‌شود. تنسورفلو مجموعه‌ای از مدل ها و الگوریتم‌های یادگیری ماشین و یادگیری عمیق است. تنسورفلو به ما این امکان را می دهد که نمودارها و ساختارهای جریان داده را بسازیم و با در نظر گرفتن ورودی‌ها به عنوان یک آرایه چند بعدی به نام تنسور، نحوه عملکرد آن ها را به صورت نمودار نشان دهیم. هم‌چنین می‌توانیم تمامی مراحلی که داده ها طی می‌کنند را به صورت فلوچارت نمایش دهیم.

معماری تنسورفلو شامل سه بخش پیش پردازش داده‌ها، ساختن مدل و آموزش مدل و تخمین عملکرد می‌باشد. این سه مرحله در کنار هم تنسورفلو را به وجود می‌آورد. تنسورفلو ورودی را به عنوان یک آرایه چند بعدی دریافت می‌کند. به این آرایه چند بعدی Tensor گفته می شود. در طی این مسیر چندین فرآیند روی داده‌های ورودی اعمال می شود و در انتها به عنوان خروجی خارج می‌شود. علت نام گذاری تنسورفلو نیز به خاطر فرآیندی است که داده ها طی می‌کنند و تانسور از طریق تعدادی عملیات مشخص جریان پیدا می‌کند و در انتها به عنوان خروجی؛ خارج می‌شود.

نام این کتابخانه از اصلی‌ترین قسمت اصلی آن یعنی Tensor گرفته شده است. تمامی محاسباتی که در تنسورفلو انجام می‌شود شامل تانسور است. اگر بخواهیم دقیق تر بگوییم که تانسور از چه چیزی تشکیل شده است، باید بگوییم که تنسور از یک ماتریس چند بعدی تشکیل شده است. این ماتریس می‌تواند هر نوع داده ای را نمایش دهد. البته این را هم باید بگوییم که درست است که تانسور قابلیت نمایش همه نوع داده را دارد اما تمامی داده‌های درون یک تانسور باید از یک نوع باشند. تانسور می تواند یک داده ورودی یا نتیجه یک محاسبه باشد. در کتابخانه تنسورفلو در پایتون تمامی عملیات روی نموداری که مجموعه ای از محاسبات متوالی است؛ انجام می‌شود. هر عملیاتی که انجام می‌شود را گره عملیات می‌گویند، تمامی گره ها به یکدیگر متصل هستند. نموداری که در طی عملیات تنسورفلو استفاده می‌شود؛ تمامی گره‌ها و اتصالات را نمایش می‌دهد. کتابخانه تنسورفلو در پایتون از نمودارها استفاده می کند. با استفاده از این نمودارها می‌توان تمام محاسبات انجام شده در طول آموزش را جمع آوری و توصیف کرد. البته مزایای گراف محدود به این موارد نیست. از دیگر مزایای گراف ها می‌توان به اجرا روی چندین CPU، GPU و حتی سیستم عامل‌های تلفن‌های همراه اشاره کرد. نمودارها قابلیت ذخیره دارند و می‌توان آن ها را برای استفاده های بعدی ذخیره کرد. تمامی محاسبات درون نمودارها به تانسور ها وابسته است و به وسیله آن‌ها انجام می‌شود. هر تانسور یک گره و یک لبه دارد. گره وظیفه انجام عملیات ریاضی را بر عهده دارد و خروجی ها را تولید می کند. لبه ها نیز وظیفه توضیح روابط ورودی و خروجی را بر عهده دارند.

الگوریتم BERT یک مدل هوش مصنوعی است که برای درک بهتر زبان طبیعی و معنای کلمات درجملات طراحی شده است. این الگوریتم از تکنیکی به نام ترانسفورمر استفاده می‌کند که به آن اجازه می‌دهد تا به صورت دو جهته متن را تحلیل کند. یعنی هم از سمت چپ به راست و هم از راست به چپ متن را بخوان و ارتباط بین کلمات را درنظر بگیرد که باعث می شود مفهوم پنهان کلمات را درک کند و نتایج دقیق تری را به کاربران ارائه دهد، به همین دلیل الگوریتم BERT نیازی به پیش پردازش متن ندارد چرا که پیش پردازش متن عملیاتی است که برای تبدیل متن به یک فرمت مناسب برای مدل های هوش مصنوعی انجام می شود. اما الگوریتم BERT می تواند متن را به صورت خام و بدون تغییر دریافت کند و با استفاده از ترانسفورمر آن را به بردارهای عددی تبدیل کند، این بردارها نشان دهنده معنای کلمات در جمله هستند و می توانند برای تشخیص احساسات استفاده شوند.

برای پیاده سازی الگوریتم BERT در این مقاله از کتابخانه Transformers استفاده می‌شود. این کتابخانه شامل ابزارهای مربوطه است. ابتدا مدل PreTrained BERT و Tokenizer که درسایت Huggingface قرار دارد و از قبل با داده‌های متنوع پیش آموزش دیده بارگزاری می‌شود و در ادامه با داده‌ها و اهداف پروژه تطبیق داده می شود.Huggingface یک پلتفرم مرکزی است که به ما امکان استفاده از جدیدترین و بهترین مدل ها و دیتاست‌های هوش مصنوعی را می‌دهد. سپس پارامترهای مدل مانند batch_size،max_lenght و تعیین می گردند، دسترسی به Gpu بررسی می‌گردد ودرصورت وجود دیتاهای موجود در هر batch به Gpu انتقال می‌یابند و یک کلاس برای Tokenizer کردن داده‌ها ساخته می‌شود. این کلاس حداکثر طول جملات ،Data و Tokenizer را می‌گیرد و دیتای موردنیاز برای ورود به مدل BERT را می‌سازد. با استفاده از Tokenizer متن نظرات به بردارهای قابل فهم برای مدل تبدیل می‌شود .بردار شامل یکسری صفر و یک است که اهمیت / عدم اهمیت هر توکن را نشان می‌دهد. متغیر attention_mask میان توکن‌های Pad و بامعنی تفاوت ایجاد می کند، صفر برای توکن‌های Pad و یک برای توکن‌های با معنی خواهد بود.

داده ها که بصورت جفت‌های متن و برچسب هستند خوانده می‌شوند و با استفاده از تابع Train-Test-Split تقسیم بندی داده‌ها برای قسمت های Test و Train و Validation انجام می‌شود. داده های آموزش برای تنظیم دقیق مدل BERT استفاده می‌شوند، داده‌های ارزیابی برای اندازه گیری عملکرد مدل BERT و داده‌های آزمون برای ارزیابی نهایی مدل بر روی داده‌های جدید استفاده می‌شوند. سپس برای بهبود عملکرد و به روز رسانی پارامترهای مدل بهینه ساز adam تعریف می‌گردد، adam سرعت زیادی دارد و حافظه کمی مصرف می‌کند.

تابع زیان cross entropy loss تعریف می‌گردد، این تابع برای مسائل دسته بندی بکار می رود. این تابع اختلاف بین دو توزیع احتمال را اندازه گیری می‌کند: توزیع احتمال پیش بینی شده توسط مدل و توزیع احتمال واقعی که برچسب های هدف را نشان می‌دهد. سپس ورودی به مدل داده شده و یک لایه خروجی مناسب اضافه می‌شود که BERT بتواند خروجی مورد نظر را تولید کند. هرلایه ترانسفورمر خروجی خود را به لایه بعدی ارسال می‌کند و درنهایت لایه آخر ترانسفورمر خروجی نهایی را تولید می‌کند. این خروجی شامل یک بردار برای هرتوکن ورودی است که معنای آن را با توجه به بافت جمله نشان می‌دهد. این بردارها می‌توانند برای انجام

وظایف مختلفی مانند طبقه بندی استفاده شوند. در مرحله بعدی مدل به حالت آموزش تنظیم می‌شود. در ابتدا مقدار اولیه خطای آموزش و دقت صفر می‌گردد و همچنین مقدار گرادیان‌ها برای جلوگیری از اختلال پاک می‌شود. سپس مدل BERT بر روی داده های آزمون اجرا شده و میزان خطا و دقت آموزش مدل محاسبه می گردد. پس از مرحله آموزش مدل وارد فاز ارزیابی می‌شود. تنظیم پارامترها مشابه فاز آموزش صورت می‌گیرد با این تفاوت که در ارزیابی به محاسبه گرادیان نیازی نیست. بعد از اجرای مدل و دریافت خروجی ها، مقدار خطا و دقت محاسبه می‌شود و نتایج چاپ می‌گردد.

پیاده سازی BERT به زبان انگلیسی هم دقیقا مشابه BERT فارسی است با این تفاوت که از یک مدل BERT که روی داده های انگلیسی Pre -Trainrd شده استفاده می‌شود. این مدل نیز از سایت Huggingface که از مدل BERT پشتیبانی می‌کند دانلود شده و بارگذاری می‌گردد و روی داده‌های آموزش داده می‌شود. سپس برای تنظیم دقیق مدل BERT یک لایه دسته بندی به انتهای مدل اضافه کرده و پارامترهای مدل با استفاده از دادههای آموزش بهینه می‌شوند و در نهایت مدل BERT روی داده‌های ارزیابی و آزمون اجرا شده و عملکرد مدل با معیارهای دقت و صحت ارزیابی می‌گردد. در مدل‌های BERT به زبان انگلیسی یا فارسی بدون نیاز به کلاس مدل‌های از پیش آموزش دیده بارگذاری می‌شوند اما برای پیاده سازی BERT دو زبانه ابتدا یک کلاس برای ساخت مدل ایجاد می‌شود سپس شخصی سازی می‌گردد. پس از ساخت کلاس مدل‌های فارسی و انگلیسی هرکدام جداگانه بارگذاری شده و یک لایه Classifier بعنوان لایه آخر معماری مدل دو زبانه تعریف می‌شود و برای خروجی لایه آخر دو مدل فارسی و انگلیسی باهم ترکیب می‌شود و مدل دو زبانه BERT ایجاد می گردد.

همانطور که شکل ۱۵ نشان می‌دهد صحت عملکرد الگوریتم BERT بر روی داده‌های این مقاله به زبان فارسی ۸۲ درصد است. پس از ترجمه داده‌ها به زبان انگلیسی با اجرای الگوریتم BERT بر روی داده‌ها صحت ۸۳ درصد بدست آمد که در شکل ۱۶ نمایش داده شده است.

	precision	recall	f1-score	support
0	0.80	0.72	0.76	79
1	0.84	0.89	0.86	126
accuracy			0.82	205
macro avg	0.82	0.81	0.81	205
weighted avg	0.82	0.82	0.82	205

شکل ۱۵- نمایی از خروجی الگوریتم BERT به زبان فارسی

	precision	recall	f1-score	support
0	0.81	0.73	0.77	79
1	0.84	0.89	0.86	126
accuracy			0.83	205
macro avg	0.82	0.81	0.82	205
weighted avg	0.83	0.83	0.83	205

شکل ۱۶- نمایی از خروجی الگوریتم BERT به زبان انگلیسی

۱-۳-۵- الگوریتم BERT دو زبانه

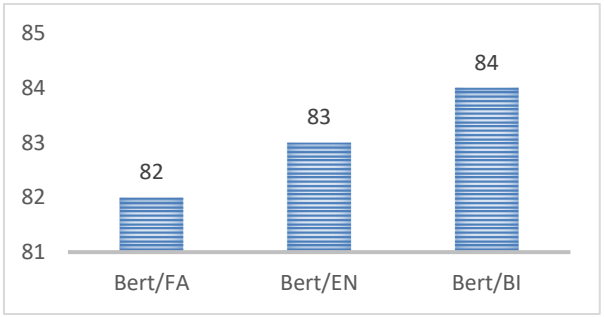
الگوریتم BERT دو زبانه یک نسخه از الگوریتم BERT است که می‌تواند با دو زبان مختلف کار کند. این الگوریتم می‌تواند متن‌هایی را که شامل دو زبان مختلف هستند، مثل فارسی و انگلیسی، تحلیل کند و معنای آن‌ها را درک کند. این الگوریتم می‌تواند برای مواردی مثل ترجمه، خلاصه‌سازی، پرسش و پاسخ و غیره مفید باشد.

الگوریتم BERT دو زبانه در سال ۲۰۱۹ توسط محققان گوگل معرفی شد و بر روی ۱۰۴ زبان مختلف آموزش داده شد. این الگوریتم از یک مدل ترنسفورمر دو طرفه استفاده می‌کند که می‌تواند با داده‌هایی که شامل دو زبان مختلف هستند، سازگار شود. برای استفاده از الگوریتم BERT دو زبانه، می‌توانیم از کتابخانه ترنسفورمرز پایتون استفاده کنید. این کتابخانه شامل مدل‌های پیش‌آموزش دیده BERT دو زبانه برای زبان‌های مختلف است. همانطور که در شکل ۱۷ نشان داده شده است صحت عملکرد این مدل برای تشخیص قطبیت نظرات کاربران سایت سهامیاب ۸۴ درصد است.

	precision	recall	f1-score	support
0	0.79	0.78	0.79	79
1	0.87	0.87	0.87	126
accuracy			0.84	205
macro avg	0.83	0.83	0.83	205
weighted avg	0.84	0.84	0.84	205

شکل ۱۷- نمایی از خروجی الگوریتمBERT دو زبانه

شکل ۱۸نشان می‌دهد که صحت اجرای الگوریتم BERT دوزبانه، بیشترین مقدار را داشته است.



شکل ۱۸- نمودار مقایسه دقت اجرای الگوریتم BERT

۶- نتایج

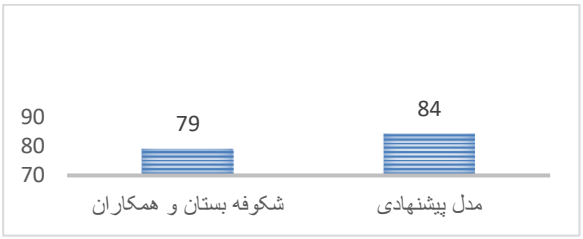
مجموعه داده‌ای که دراین مقاله ارائه شده دارای ۱۰۲۴ نمونه است. هرنمونه از یک نظر به همراه برچسب آن (مثبت یا منفی) و نام سهام شرکت مورد نظر تشکیل شده است. در ابتدا مقایسه الگوریتم های سنتی یادگیری ماشین و عملکرد آن‌ها بررسی شد. برای این منظور الگوریتم های درخت تصمیم، ماشین بردار پشتیبان، بیزین ساده وk نزدیک‌ترین همسایگی (KNN) پیاده سازی شدند. بردارپشتیبان با صحت ۶۱ درصد بهترین عملکرد را درمیان الگوریتم های سنتی به دست آورد. درفاز بعدی به منظور مقایسه الگوریتم های یادگیری سنتی با الگوریتم های یادگیری عمیق مدل‌های LSTM و BERT به زبان فارسی و BERT به زبان انگلیسی طراحی شدند. این مدل ها به ترتیب با صحت های ۷۲ درصد ، ۸۲ درصد و ۸۳ درصد عملکرد بهتری نسبت به الگوریتم های سنتی داشتند. درمرحله بعدی مدل LSTM با استفاده از الگوریتم فرا ابتکاری ژنتیک باهدف بدست آوردن هایپر پارامترهای بهینه پیاده سازی شد. باکمک الگوریتم ژنتیک این مدل به صحتی برابر با ۸۱/۴۶ درصد رسید که ۹/۴۶درصد نسبت به مدل اولیه LSTM پیشرفت داشت. در فاز پایانی این مقاله الگوریتم BERT دو زبانه با ترکیب متن فارسی نظرات و معنای

انگلیسی آنها پیاده سازی شد و به عملکرد ۸۴ درصد دست یافت. دقت حاصل از این پژوهش نسبت به کار انجام شده توسط شکوفه بستان و همکاران [۱] به عنوان مدل پایه در این مقاله در نظر گرفته شده است به میزان ۵ درصد افزایش داشته که نسبتا قابل توجه است. جدول ۴ مقایسه دقت مدل پیشنهادی و مدل پایه (کار انجام شده توسط بستان و همکاران) را نمایش می‌دهد.

جدول ۴- مقایسه دقت مدل پیشنهادی با مدل پایه

مدل	مدل پایه	مدل پیشنهادی
BERT چند زبانه دوکلاسه	۷۹ درصد	۸۴ درصد

همانطورکه شکل ۱۹ نشان می‌دهد دقت حاصل از این پژوهش نسبت به کار انجام شده توسط شکوفه بستان و همکاران به میزان ۵ درصد افزایش داشته است.



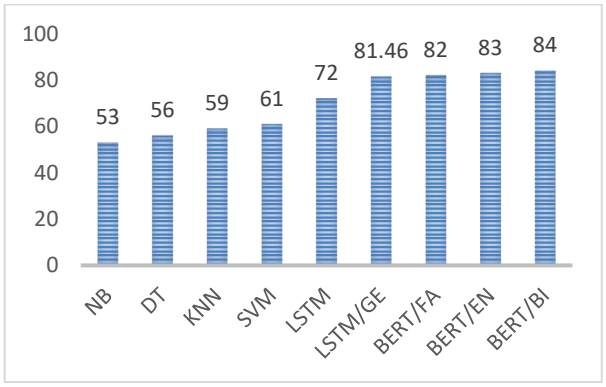
شکل ۱۹- مقایسه دقت مدل پیشنهادی و مدل پایه (کار انجام شده توسط بستان و همکاران [۱])

جدول ۵ نتایج اجرای هرکدام از این مدل‌ها را نشان می‌دهد. مشاهده می‌شود که مدل عمیق چندزبانه BERT بهترین عملکرد را داراست.

جدول ۵- نتایج اجرا

نوع زبان	مدل	برچسب (کلاس)	صحت
فارسی	SVM	مثبت	۶۴
		منفی	۵۰
		مجموع	۶۱
فارسی	Decision tree	مثبت	۶۶
		منفی	۴۴
		مجموع	۵۶
فارسی	Naïve Bayes	مثبت	۶۲
		منفی	۴۰
		مجموع	۵۳
فارسی	KNN	مثبت	۶۳
		منفی	۴۵
		مجموع	۵۹
فارسی	LSTM	مثبت	۷۶
		منفی	۶۵
		مجموع	۷۲
فارسی	Genetic + LSTM	مثبت	۸۴
		منفی	۷۳
		مجموع	۸۱/۴۶
فارسی	BERT	مثبت	۸۴
		منفی	۸۰
		مجموع	۸۲
انگلیسی	BERT	مثبت	۸۴
		منفی	۸۱
		مجموع	۸۳
دوزبانه	BERT	مثبت	۸۷
		منفی	۷۹
		مجموع	۸۴

همانگونه که در شکل ۲۰ دیده می‌شود این مدل با صحت ۸۴ درصد عملکرد مناسبی برای تحلیل نظرات کاربران سایت سهامیاب ارائه می دهد.



شکل ۲۰- نمودار نتایج اجرا

۷- نتیجه گیری

هدف این مقاله ارائه مدلی مبتنی بر یادگیری عمیق به منظور تحلیل نظرات کاربران در خصوص سهام چند شرکت فعال بورس است که از وب سایت سهامیاب جمع آوری شده‌اند. تحلیل این نظرات با بهره گیری از الگوریتم‌های سنتی و الگوریتم های یادگیری عمیق با دو زبان فارسی و انگلیسی پیاده‌سازی می‌گردد. برای مقایسه الگوریتم‌های یادگیری سنتی با الگوریتم‌های یادگیری عمیق مدل‌های LSTM و BERT به زبان فارسی و BERT به زبان انگلیسی طراحی شدند. پس از پیاده سازی مدل‌ها، ماشین بردار پشتیبان بهترین عملکرد را در میان الگوریتم های سنتی بدست آورد و مدل های یادگیر عمیق عملکرد بهتری نسبت به الگوریتم های سنتی داشتند. در ادامه مدل LSTM بااستفاده از الگوریتم فرابتکاری ژنتیک با هدف به دست آوردن هایپر پارامترهای بهینه پیاده سازی شد. باکمک الگوریتم ژنتیک این مدل نسبت به مدل اولیه LSTM پیشرفت داشت. در فاز پایانی این پژوهش الگوریتم BERT دو زبانه با ترکیب متن فارسی نظرات و معنای انگلیسی آن‌ها پیاده سازی شد که به عملکرد ۸۴ درصد دست یافت.

انتظار می رود این مدل عمیق چند زبانه نقش قابل توجهی در بهبود عملکرد سیستم های توصیه گر در این حوزه ایفا نماید. با توجه به اهمیت حجم دیتا در فرایند آموزش مدل‌های مبتنی بر شبکه‌های عصبی عمیق، می‌توان قبل از عملیات Fine-tune، مدل‌های زبانی را بر روی حجم زیادی از دیتای بدون برچسب در این حوزه آموزش داد که کمک شایانی به درک مدل‌های زبانی از ادبیات حوزه مذکور خواهد نمود. بدیهی است که این امر نیازمند جمع‌آوری داده در حجم بالا می‌باشد. همچنین با افزایش دیتای برچسب‌زده، فرایند تشخیص احساسات بهبود خواهد یافت. علاوه بر این، در صورت فراهم نمودن منابع سخت‌افزاری مورد نیاز مدل‌های زبانی بزرگ (Large Language Models) یا به اختصار LLMs، می‌توان از قدرت بالای یادگیری و پاسخگویی این معماری نیز بهره گرفت. یکی دیگر از کارهایی که می‌توان در آینده نسبت به پیاده‌سازی آن اقدام نمود، ایجاد یک API با هدف تحلیل احساسات Realtime و ارائه پیشنهاد برخط به معامله‌گران این حوزه می‌باشد. همچنین با توسعه این سیستم، امکان ایجاد دستیار معامله با هدف انتقال دانش و تحلیل روانشناسی در کوتاه‌ترین زمان ممکن فراهم خواهد آمد.

۸- مراجع

[۱] بستان شکوفه، زارع بیدکی علی محمد، پژوهان محمدرضا، "درون سازی معنایی واژه ها با استفاده از BERT روی وب فارسی"، *مهندسی برق و مهندسی کامپیوتر ایران* - ب مهندسی کامپیوتر، شماره ۲، دوره ۲۱، شماره:۲، صفحات ۸۹-۱۰۰، ۱۴۰۲.

[2] Antonakaki, D., Fragopoulou, P., & Ioannidis, S., A survey of Twitter research: Data model, graph structure, sentiment analysis and attacks. *Expert Systems with Applications*, 164, 114006, 2021.

[3] Basiri, M. E., Nemati, S., Abdar, M., Cambria, E., & Acharya, U. R., ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis, *Future Generation Computer Systems*, 115, 279-294, 2021.

[4] Dashtipour, K., Gogate, M., Cambria, E., & Hussain, A., A novel context-aware multimodal framework for persian sentiment analysis, *Neurocomputing*, 457, 377-388, 2021.

[5] Eck, M., Germani, J., Sharma, N., Seitz, J., & Ramdasi, P. P., Prediction of stock market performance based on financial news articles and their classification, Data Management, *Analytics and Innovation*, vol 1175, pp. 35-44, 2020.

[6] Hong, S., A study on stock price prediction system based on text mining method using LSTM and stock market news, *Journal of Digital Convergence*, 18(7), 223-228, 2020.

[7] Li, H., Chen, Q., Zhong, Z., Gong, R., & Han, G., E-word of mouth sentiment analysis for user behavior studies, *Information Processing & Management*, 59(1), 102784, 2020.

[8] Li, Y., & Pan, Y., A novel ensemble deep learning model for stock prediction based on stock prices and news*, International Journal of Data Science and Analytics*, Volume 13, pages 139–149, 2021.

[9] Liu, B., Sentiment analysis: Mining opinions, sentiments, and emotions, *Cambridge university press*, 2020.

[10] Lutz, B., Pröllochs, N., & Neumann, D., Predicting sentence-level polarity labels of financial news using abnormal stock returns, *Expert Systems with Applications*, 148, 113223, 2020.

[11] Man, R., & Lin, K., Sentiment analysis algorithm based on BERT and convolutional neural network, *In 2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)* (pp. 769-772). IEEE, 2021.

[12] Mate, G. S., Kulkarni, R., Amidwar, S., & Muthya, Stock prediction through news sentiment analysis, *Journal of Architecture & Technology*, 11(8). 36-40, 2020.

[13] Mitra, A., Sentiment analysis using machine learning approaches (Lexicon based on movie review dataset), *Journal of Ubiquitous Computing and Communication Technologies (UCCT),* 2(03), 145-152, 2020.

[14] Nemati, S., Rohani, R., Basiri, M. E., Abdar, M., Yen, N. Y., & Makarenkov, V., A hybrid latent space data fusion method for multimodal emotion recognition, *IEEE Access*, 7, 172948-172964, 2019.

[15] Sadr, H., Pedram, M. M., & Teshnehlab, M., Multi-view deep network: a deep model based on learning features from heterogeneous neural networks for sentiment analysis, *IEEE access*, 8, 86984-86997, 2020.

[16] Sallam, R. M., Hussein, M., & Mousa, H. M., Improving collaborative filtering using lexicon-based sentiment analysis, *International Journal of Electrical and Computer Engineering*, 12(2), 1744, 2022.

[17] Shah, A. M., Yan, X., Shah, S. A. A., & Mamirkulova, G. Mining patient opinion to evaluate the service quality in healthcare: a deep-learning approach, *Journal of Ambient Intelligence and Humanized Computing*, 11(7), 2925-2942, 2020.

[18] Shah, P., Swaminarayan, P., & Patel, M., Sentiment analysis on film review in Gujarati language using machine learning, *International Journal of Electrical and Computer Engineering*, 12(1), 1030, 2022.