

## یادگیری عمیق در تحلیل احساسات کاربران شبکه های هوشمند سرمایه گذاری

نسرین صالحی چگنی<sup>۱</sup>، صبا جودکی<sup>۲\*</sup>، مجتبی صالحی<sup>۳</sup>

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

<sup>۱</sup>گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران

<sup>۲\*</sup>گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران (saba.joudaki@iau.ac.ir)

<sup>۳</sup>گروه مهندسی کامپیوتر، واحد خرم آباد، دانشگاه آزاد اسلامی، خرم آباد، ایران

### چکیده

تحلیل داده های سهام شرکت ها یکی از روش های مهم برای ارزیابی عملکرد و ارزش شرکت ها و انتخاب بهترین گزینه برای سرمایه گذاری در بازارهای مالی است. در این مقاله، مجموعه داده نظرات کاربران شبکه هوشمند سرمایه گذاری سهام یاب با هدف تحلیل احساسات کاربران گردآوری شده است. ابتدا الگوریتم های درخت تصمیم، ماشین بردار پشتیبان، بیزین ساده و نزدیکترین همسایه پیاده سازی شدند؛ که بردار پشتیبان با صحت ۶۱ درصد بهترین عملکرد را به دست آورد. سپس برای مقایسه این الگوریتم های یادگیری سنتی با الگوریتم های یادگیری عمیق، شبکه های 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 BERT دو زبانه با ترکیب متن فارسی نظرات و معنای انگلیسی آن ها پیاده سازی شد و به عملکرد ۸۴ درصد دست یافت. انتظار می رود استفاده از این مدل بتواند به بهبود عملکرد سیستم های پیش بینی کننده و توصیه گر در سایت های اقتصادی کمک نماید.

**کلمات کلیدی:** پردازش زبان طبیعی، تحلیل احساسات، یادگیری عمیق، شبکه عصبی، سیستم های توصیه گر

### ۱- مقدمه

از سویی دیگر، روش های سنتی در نادیده گرفتن اطلاعات معنایی مندرج در متن محدودیت دارند. یکی دیگر از مسایل مطرح در روش های 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 روش های یادگیری عمیق (Deep Learning) نسبت به روش های سنتی یادگیری ماشین از اولویت بالاتری برخوردار می باشد. یادگیری عمیق شاخه ای از یادگیری ماشین است. به عبارت دیگر همان یادگیری به وسیله شبکه های عصبی است که لایه های مخفی (Hidden Layers) زیادی دارد. روش های یادگیری ماشین در دو دسته اصلی با ناظر در صورت وجود برچسب و بدون ناظر در حالت بدون برچسب تقسیم می شوند که بیش تر کارهای انجام شده در سطح تحلیل متن مبتنی بر روش های با ناظر می باشند. بازار مالی به عنوان یک سیستم پیچیده با نوسانات غیر خطی از جمله بخش هایی است که از مزایای الگوریتم های 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

تجزیه و تحلیل احساسات یا نظر کاوی یک حوزه در حال پیشرفت است که به استفاده از پردازش زبان طبیعی، تجزیه و تحلیل متن اشاره دارد و برای استخراج کمیت مورد استفاده قرار می گیرد و برای مطالعه حالات احساسی از یک بخش مشخص از اطلاعات یا مجموعه داده های متنی مورد استفاده قرار می گیرد. تحلیل احساسات در بسیاری از شرکت ها برای بررسی محصولات، نظرات رسانه های اجتماعی و برای بررسی مثبت، منفی یا خنثی بودن متن استفاده می شود. بیشتر مطالعات در زمینه تحلیل احساسات در زبان انگلیسی از روش های یادگیری ماشین (Machine Learning) سنتی مانند SVM، درخت تصمیم گیری و Naive Bayes استفاده می کنند. این روش ها معمولاً از ویژگی های n-gram و لغوی بهره می برند. در زبان شناسی مفهوم n-gram به معنای دنباله ای پیوسته از n جزء در یک دنباله معین از متون است. این اجزا می توانند حروف، واج، هجا یا واژه باشند.

ویژگی های محلی مقاوم در برابر موقعیت، ABCDM از مکانیزم های کانولوشن و ادغام استفاده می کند. اثربخشی ABCDM بر تشخیص قطبیت احساسی که رایج ترین و اساسی ترین وظیفه تحلیل احساسی است، ارزیابی می شود. آزمایش ها در پنج بررسی و سه مجموعه داده توپیت انجام شده است. نتایج مقایسه ABCDM با شش DNN که اخیراً برای تجزیه و تحلیل احساسی پیشنهاد شده اند نشان می دهد که ABCDM به نتایج بسیار خوبی در بررسی بلند مدت و طبقه بندی قطبیت کوتاه توپیت دست می یابد.

دستی پور و همکاران [۴] یک مجموعه داده چند وجهی فارسی شامل بیش از ۸۰۰ جمله را به عنوان 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 نشانه های صوتی، بصری و متنی برای تعیین دقیق تر احساسات بیان شده بهره می برد. از هر دو روش ترکیب سطح تصمیم (دیرنگام) و سطح ویژگی (زود هنگام) برای ادغام اطلاعات موثر متقابل استفاده کردند. نتایج تجربی نشان می دهند که ادغام بافتی ویژگی های چند وجهی مانند ویژگی های متنی، صوتی و بصری عملکرد بهتری (۹۱.۳۹٪) را در مقایسه با ویژگی های تک وجهی (۸۹.۲۴٪) ارائه می دهند.

آک و همکاران [۵] از اخبار مالی برای پیش بینی عملکرد بازار سهام استفاده کردند. هنگ [۶] یک سیستم پیش بینی 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 داده های تحلیل سری زمانی گذشته، نزدیک ترین 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 سایت های همسریابی به عنوان یک نمونه انتخاب کردند و هنگام تست و مقایسه کارایی تحقیق رفتار کاربر، از تحلیل احساسی مختلف، تکنیک های یادگیری ماشین و تحلیل احساسی مبتنی بر فرهنگ لغت استفاده کردند و دریافتند که ترکیب یادگیری ماشین و روش مبتنی بر واژگان می تواند به دقت بالاتری نسبت به هر نوع تحلیل احساسی دست یابد.

در تحقیقی دیگر با ایجاد سیستم های جدید تحلیل گر احساسات بر مبنای تکنیک های مختلفی از شبکه عصبی عمیق، دقت پیش بینی 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 پیش بینی برچسب های قطبیت سطح جمله در اخبار مالی توسعه دادند. روش آن ها از نمایش متن توزیع شده و یادگیری چند نمونه ای برای انتقال اطلاعات از سطح سند به سطح جمله استفاده کرد. سیستم خبره پیشنهادی می تواند به سرمایه گذاران در تصمیم گیری خود کمک کند و ممکن است به آن ها در برقراری ارتباط با پیام های مورد نظر کمک کند.

روبیمن و همکاران [۱۱] یک الگوریتم تجزیه و تحلیل احساسی متن چینی را بر اساس BERT و CNN پیشنهاد می کنند. این شبکه از "BERT" برای استخراج ویژگی های هر کلمه و استفاده از آن به عنوان ورودی سی ان ان استفاده می کند. آزمایش ها نشان می دهند که این مدل از نظر اثربخشی امکان پذیر است. مدل BERT ساختار Encode - Decoder را براساس چارچوب ترانسفورمر پیاده سازی می کند. ساختار ترانسفورمر در ترانسفورماتور هم چنین از چارچوب Encoder - Decoder استفاده می کند 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 پیش خور است. توجه به خود به گره فعلی اجازه می دهد 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 پیش خور است. لایه

بینی سود سهام، پیش بینی بازار بورس، پیش بینی نرخ ارز، پیش بینی ورشکستگی بانک ها، تحلیل احساسات سرمایه گذاران، تحلیل نظرات مشتریان و تشخیص کلاه برداران مالی به ایفای نقش پرداخته است. عموماً سرمایه گذاران بازار مالی به منظور سود بیش تر، نظرات و حالت های احساسی خود را با دیگر سرمایه گذاران به اشتراک می گذارند و در نتیجه رفتار سرمایه گذاران مالی بر قیمت های بازار سهام تاثیرگذار خواهد بود. ولی از آن جایی که غالب سرمایه گذاران از روند تغییرات بازارهای مالی اطلاع ندارند، بررسی حالت های احساسی سرمایه گذاران و شناسایی الگوهای مخفی بین نظرات مردم در رسانه های اجتماعی، می تواند آن ها را برای رسیدن به سودآوری بیش تر، شناخت تغییرات بازار و پیش گویی حوادث اقتصادی یاری کند. امروزه محققان در صدد هستند تا با به کارگیری تکنیک های هوش مصنوعی مانند شبکه های عصبی پیچشی و یا سایر نمونه های شبکه های عصبی عمیق به ارزیابی نظرات، تحلیل داده های احساسی مشتریان یا کاربران در فضاهای مجازی و نیز شناسایی و پیش بینی تغییرات در بازار بپردازند. سیستم های هوشمند تحلیل گر احساسات، نقش مهمی را در تحلیل و طبقه بندی نظرات، توصیف ها و نگرش های مردم نسبت به موضوعات مطرح در کامنت های شبکه های اجتماعی بازی می کنند.

هدف این مقاله ارائه یک مدل مبتنی بر شبکه عصبی عمیق به منظور تحلیل نظرات کاربران سایت سهامیاب و مقایسه دقت و کارایی مدل پیشنهادی با مدل های سنتی یادگیری ماشین است. برای این منظور، نظرات کاربران سایت سهامیاب در خصوص بورس تهران جمع آوری شد. سپس بار معنایی جملات با استفاده از تکنیک های متن کاوی و آنالیز احساسات تعیین گردید و با استفاده از الگوریتم های یادگیری ماشینی به دسته های مثبت و منفی طبقه بندی شد به این شکل که نظرات مثبت با عدد ۱ و نظرات منفی با عدد صفر برچسب گذاری شدند.

از آن جا که متن، داده ای بدون ساختار است، پیش پردازش برای تبدیل این داده های بدون ساختار به فرم ساختار یافته مورد نیاز است. سپس با انجام فرآیند یادگیری با استفاده از داده های مجموعه آموزشی (Train) الگوریتم های ماشین یادگیر پیاده سازی شده و مدل یادگیری ماشین ساخته می شود و در نهایت با استفاده از داده های آزمایشی (Test) ارزیابی می شود. در حالت کلی مراحل طراحی و پیاده سازی به صورت جمع آوری مجموعه داده، برچسب گذاری داده ها، پیش پردازش داده، تشکیل بردار کلمه، اجرای الگوریتم های سنتی یادگیری ماشین، طراحی شبکه عصبی، اجرای الگوریتم ژنتیک، ارزیابی کلی نتایج و ارائه مدل نهایی بیان می شود.

## ۲ - مطالعات انجام شده

بستان و همکاران [۱] درون سازی واژگان فارسی با استفاده از الگوریتم BERT 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 قرار گرفت 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 مدل های مورد بررسی داشت و دقت را نسبت به مدل BERT چندزبانه تا حداقل یک درصد افزایش داد.

انتونا کاکو و همکاران [۲] موضوعات تحقیقاتی فعلی را در توپیت با تمرکز بر سه حوزه اصلی ترسیم کردند: ساختار و ویژگی های نمودار اجتماعی، تحلیل احساسی و تهدیدهایی مانند اسپم، ربات ها، اخبار جعلی و سخنان نفرت آمیز. همچنین مدل داده پایه توپیت و بهترین روش ها برای نمونه گیری و دسترسی به داده را ارائه دادند و زمینه تکنیک های محاسباتی مورد استفاده در این حوزه ها مانند نمونه برداری گراف، پردازش زبان طبیعی و یادگیری ماشینی را ارائه دادند.

بصری و همکاران [۳] اولین مدل عمیق دو جهتی CNN - RNN مبتنی بر توجه را برای تحلیل احساسی پیشنهاد کردند. مدل آن ها هم در طبقه بندی طولانی و هم در طبقه بندی کوتاه قطبیت توپیت به نتایج سطح بالا دست یافت. همچنین، مکانیزم توجه بر روی خروجی های لایه های دو طرفه ABCDM برای تأکید بیش تر یا کم تر بر روی کلمات مختلف اعمال می شود. برای کاهش ابعاد ویژگی ها و استخراج

جدول ۱- نمای کلی مجموعه داده

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

### ۲-۳- برچسب گذاری مجموعه داده ها

برچسب گذاری مجموعه داده فرآیندی در یادگیری ماشینی است که در آن داده های خام مانند تصاویر، فایل های متنی، ویدئوها و غیره را می توان شناسایی کرد و برای ارائه زمینه ای که اجازه می دهد یک یا چند برچسب معنی دار و آموزنده را اضافه کرد، استفاده می شود. به طوری که مدل یادگیری ماشینی بتواند چیزی از آن بیاموزد؛ هم چنین اجازه می دهد تا یک مجموعه داده را در یادگیری ماشینی برچسب گذاری کنید و در یادگیری نظارت شده، برچسب گذاری مجموعه داده، بخش مهمی از پیش پردازش داده است؛ بنابراین برای طبقه بندی می تواند ورودی و خروجی را برچسب گذاری کند. برچسب گذاری داده ها فرآیند مهمی است؛ زیرا می تواند قبل از استفاده از آن در مدل آموزشی، زمینه و مفهوم را به داده ها اضافه کند، به طوری که برچسب گذاری داده ها به ما کمک می کند تا زمانی که می خواهیم عامل مقیاس پذیری و فاکتور کیفیت را بهبود بخشیم، رویکرد صحیحی را انتخاب کنیم. برچسب گذاری داده ها فرآیند شناسایی داده های خام و برچسب گذاری 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 هر کدام از این برچسب ها قطبیت نظرات در مورد شرکت های فعال بورس را بیان می کند.

### ۳-۳- پیش پردازش داده

با توجه به اینکه داده هایی که در این مقاله استفاده شده است از بستر اینترنت جمع آوری شده اند، ساختار مناسبی برای ورود به شبکه های عصبی عمیق ندارند، بنابراین قبل از ورود به سیستم باید به شکلی قابل فهم برای الگوریتم ها تبدیل شوند. در همین راستا تغییراتی در قالب پیش پردازش بر روی این داده ها اعمال می شود. از آن جایی که داده های 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 پیش پردازش بر روی این داده ها ضروری است.

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

یکی از کتابخانه های اصلی پایتون برای آماده سازی و پیش پردازش داده ها، پانداس (Pandas) است که کارایی بالا، ساختاری با قابلیت استفاده آسان و ابزارهای تحلیل داده برای زبان برنامه نویسی پایتون را فراهم می کند. توابع گوناگون Pandas به ساده سازی فرآیند پیش پردازش داده ها کمک قابل توجهی می کنند. در واقع، می توان گفت پانداس یک کتابخانه قدرتمند برای تحلیل، و پیش پردازش (PreProcessing) داده ها است. این کتابخانه می تواند داده ها را با بهره گیری از ساختارهای Series و DataFrame که ارائه می کند، به قالبی که برای تحلیل داده ها مناسب هستند، مبدل سازد.

توجه می تواند 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 پیش بینی استفاده کرد. آن ها هم چنین 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 وابستگی های طولانی مدت را استخراج کرده و زبان اطلاعات محلی را کاهش دهند. آن ها 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 می کنند. آن ها ادعان داشتند که شبکه های عصبی ژرف مختلف به دلیل ساختارهای متمایز قادر به استخراج انواع مختلف ویژگی ها هستند، بنابراین در مدل پیشنهادی، ویژگی های استخراج شده از شبکه های عصبی ناهمگن را با استفاده از طبقه بندی کننده های چندنمایی ترکیب کردند تا با در نظر گرفتن ارتباط بین آن ها، بتوانند عملکرد تحلیل گر 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 توصیه هایی برای کتاب ها پیشنهاد کرده است. 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 داده است.

در تحقیقات دیگری تحلیل احساسات بیماران [۱۷] و نیز مشاهده کنندگان فیلم ها [۱۸] در سه مدل یادگیری عمیق حافظه کوتاه مدت طولانی (LSTM)، شبکه های عصبی عمیق (DNN) و شبکه های 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.

#### ۱-۳- آماده سازی داده ها

قدم اول در پیاده سازی مدل پیشنهادی آماده سازی داده ها است. آماده کردن داده های ورودی یکی از مهم ترین مراحل پیاده سازی الگوریتم های یادگیری ماشین است. ابتدا برچسب گذاری نظرات جمع آوری شده از سایت سهامیاب (www.sahamyab.com) انجام می شود. سپس معادل انگلیسی نظرات با استفاده از ابزار مترجم گوگل ایجاد می گردد. در آخرین مرحله نظرات انگلیسی و فارسی و برچسب های متناظر با هر کدام از این نظرات در یک فایل اکسل با نام sahamyab\_dataset ذخیره می شوند.

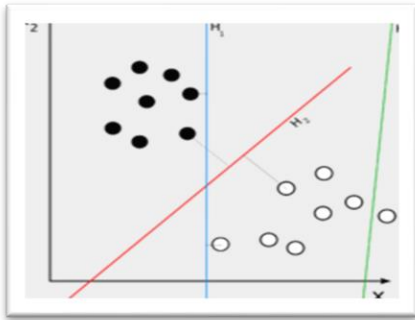
معمولاً برای کار با داده هایی که در یک فایل اکسل ذخیره شده اند به کتابخانه pandas نیاز است. خوشبختانه، پانداس متدهای زیادی را فراهم می کند که می توان از آن ها برای بارگذاری داده ها از چنین منابعی در دیتافریم پانداس استفاده کرد. پانداس از تابع read\_excel نیز استفاده می کند که می تواند برای خواندن داده های Excel در یک دیتافریم پانداس استفاده شود.

مجموعه داده ای که در این پژوهش استفاده می شود شامل نظرات کاربران سایت سهامیاب در خصوص سهام چند شرکت فعال بورس تهران است که از وب سایت سهامیاب جمع آوری شده اند. در جدول ۱ نمای کلی این مجموعه داده آمده است.

روش های نسبتاً جدیدی است که در سال های اخیر کارایی خوبی نسبت به روش های قدیمی تر برای طبقه بندی نشان داده است. پیش از آن که در صحت و بررسی در مورد ماشین های 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 شکل ۱ H3 (سبز) دو دسته را از هم جدا نمی کند اما H1 (آبی) دو دسته را از هم با حاشیه ای کوچک جدا می کند و H2 (قرمز) دو دسته را با حداکثر حاشیه از هم جدا می سازد.



شکل ۱- خط یا ابر صفحه جداکننده

#### ۴-۱-۲- حداکثر حاشیه

بر طبق قضیه ای در تئوری یادگیری اگر مثالی آموزشی به درستی دسته بندی شده باشند، از بین جداسازهای خطی، آن جداسازی که حاشیه داده های آموزشی را حداکثر می کند، خطای تعمیم را حداقل خواهد کرد. این روش به نظر مطمئن ترین راه است و به طور تجربی به خوبی جواب داده و البته تئوری هایی بر مبنای VC dimension وجود دارد که مفید بودن آن را اثبات می کند. دلیل این که SVM روی بزرگترین مرز برای Hyperplane پافشاری می کند این است که قضیه قابلیت عمومیت بخشیدن به الگوریتم را بهتر تامین می کند. این نه تنها به کارایی طبقه بندی و دقت آن روی داده های آزمایشی کمک می کند، فضا را نیز برای طبقه بندی بهتر داده های آتی مهیا می کند. به طور حسی آن مرزی که به صورت بخشی از فضا تعریف می شود یا همان تفکیک بین دو کلاس به وسیله Hyperplane تعریف می شود. همین تعریف هندسی به ما اجازه می دهد تا کشف کنیم که چگونه مرزها را بیشینه کنیم ولو این که تعداد بیشمار Hyperplane داشته باشیم و فقط تعداد کمی، شایستگی راه حل برای SVM دارند.

#### ۴-۱-۳- بردار پشتیبان

شکل ۲ ابر صفحه ای با حداکثر حاشیه برای یک ماشین بردار پشتیبان که با نمونه داده هایی از دو دسته یاد گرفته شده است را نمایش می دهد. داده هایی که بر روی ابر صفحه حاشیه قرار دارند بردارهای پشتیبان نام دارند به عبارتی نزدیک ترین داده های آموزشی به ابر صفحه های جداکننده بردار پشتیبان نامیده می شوند.

یک Series مشابه با آرایه یک بُعدی است. Series می تواند داده ها از هر نوعی را ذخیره کند. مقادیری که در Series قرار می گیرند قابل تغییر هستند؛ اما اندازه Series پانداس، غیر قابل تغییر است. به اولین عنصر در Series، اندیس صفر تخصیص داده می شود 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 می توان به عنوان یک جدول در نظر گرفت. دیتافریم، داده ها را در سطرها و ستون ها سازمان دهی می کند و از آن ها یک ساختار داده دو بُعدی می سازد. ستون ها می توانند حاوی مقادیری از انواع گوناگون باشند و در عین حال، اندازه دیتافریم قابل تغییر است؛ بنابراین می توان آن را ویرایش کرد. برای ساخت دیتافریم، می توان کار را از پایه شروع کرد و یا ساختار داده هایی مانند آرایه های (Numpy) را به یک دیتافریم تبدیل کرد. ساختاری که در اینجا استفاده شده است ساختار DataFrame است.

پس از این که 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 بر حسب متناظر با هر کدام از این نظرات است. ابتدا با استفاده از کتابخانه Pandas فایل مجموعه داده خوانده و آماده پیش پردازش می شود. به این 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 لغت نامه از کلمات موجود ساخته می شود و بعد از این که عملیات نرمال سازی روی این کلمات صورت گرفت آماده ورود به الگوریتم های یادگیری می شوند.

از آن جایی که ورودی شبکه های عصبی به صورت عدد است، لغت نامه ای که در این پژوهش استفاده می شود با استفاده از اندیس متناظر کلمات ایجاد می شود. به این منظور برای هر کدام از لغات یک اندیس عددی در نظر گرفته می شود. هر نظر با لیستی از اعداد صحیح بیان می شود. باتوجه به این که طول داده های ورودی شبکه های عصبی باید یکسان باشد با استفاده از تابع Pad-Sequence از کتابخانه Keras عملیات نرمال سازی روی نظرات انجام می شود و طول تمام نظرات یکسان و برابر عدد ۳۰ می شود.

مرحله آخر تقسیم داده ها به سه دسته آموزش، اعتبارسنجی و آزمون است. در این پژوهش داده ها به نسبت های ۶۴ درصد، ۱۶ درصد و ۲۰ درصد برای هر کدام از دسته های آموزش، اعتبارسنجی و آزمون تقسیم شده است. به این صورت که با استفاده از تابع Train\_Test\_Split از کتابخانه sklearn، ۲۰ درصد داده ها برای آزمون (Test) و ۸۰ درصد باقیمانده با نسبت ۸۰ به ۲۰ برای آموزش (Train) و اعتبارسنجی (Validation) تقسیم بندی شده اند.

## ۴- الگوریتم های سنتی یادگیری ماشین

یکی از اهداف این پژوهش مقایسه مدل پیشنهادی خود با مدل های سنتی یادگیری ماشین جهت بررسی دقت و کارایی است. برای بررسی این هدف تعدادی از الگوریتم های سنتی یادگیری ماشین مانند SVM، KNN، درخت تصمیم و بیزین ساده با استفاده از کتابخانه Sklearn پیاده سازی می شوند. Scikit Learn از کتابخانه های متن باز، مفید، پر کاربرد و قدرتمند در زبان برنامه نویسی پایتون است که برای اهداف یادگیری ماشین به کار می رود. این کتابخانه ابزارهای کاربردی زیادی به منظور یادگیری ماشین و مدل سازی آماری داده ها هم چون طبقه بندی (Classification)، رگرسیون، خوشه بندی و کاهش ابعاد فراهم می کند. این کتابخانه که به طور عمده توسط زبان پایتون ارائه شده، بر پایه ی کتابخانه های Scipy، Numpy و Matplotlib طراحی شده است.

#### ۴-۱- ماشین بردار پشتیبان

ماشین بردار (Support Vector Machin) یکی از روش های یادگیری با نظرات است که از آن برای طبقه بندی و رگرسیون استفاده می کنند. این روش از

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 پیش فرض rbf است.

همانطور که شکل ۳ نشان می‌دهد 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- نمایشی 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

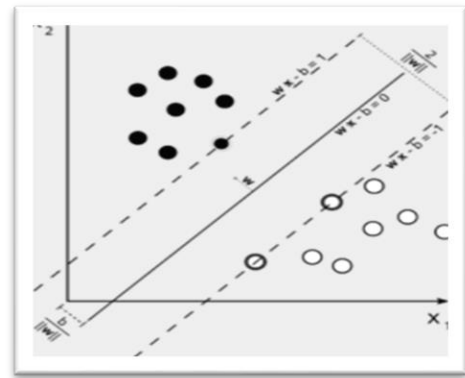
کاذب یا ( False Positive )

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



شکل ۲- بردار پشتیبان

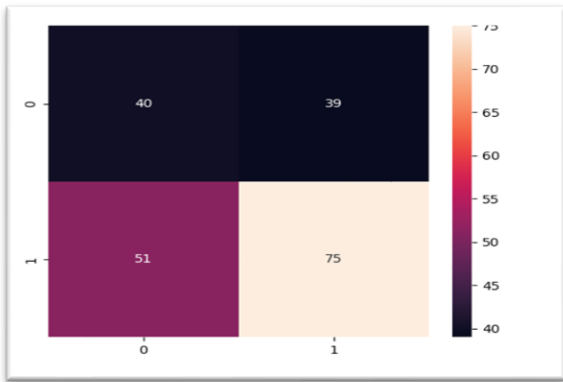
#### ۴-۱-۴- ماشین بردار پشتیبان خطی

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 ( $-1$  یا  $+1$ ) 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 این که ماشین بتواند داده‌های با پیچیدگی بالا را دسته‌بندی کند داده‌ها را با استفاده از تابع  $\Phi$  به فضای با ابعاد خیلی بالاتر می‌بریم. برای این که بتوانیم مسئله ابعاد خیلی بالا را با استفاده از این روش‌ها حل کنیم از قضیه دوگانگی لاگرانژ برای تبدیل مسئله مینیمم‌سازی مورد نظر به فرم دوگانگی آن که در آن به جای تابع پیچیده  $\Phi$  که ما را به فضای 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 نامی، چند جمله‌ای و سیگموئید می‌توان استفاده نمود. با فرض این که 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 نگاهت 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 در فضای داده‌ها به بهترین شکل ممکن تفکیک داده‌ها را انجام می‌دهد. 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 می‌گیرد. SVC طبقه‌بندی



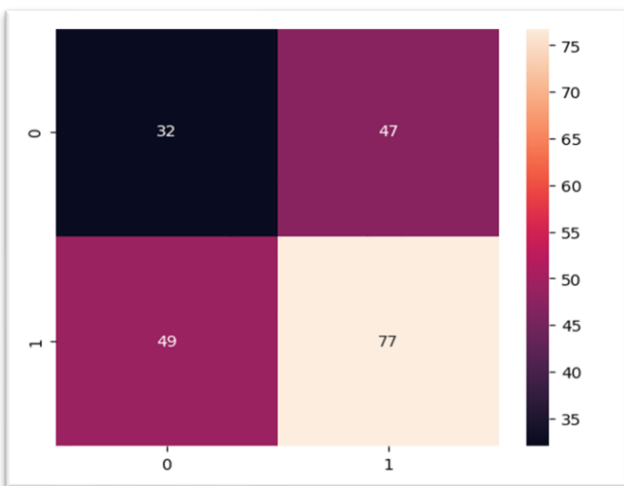
شکل ۶- نمودار Heat Map الگوریتم درخت تصمیم

### 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. همانطور که 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

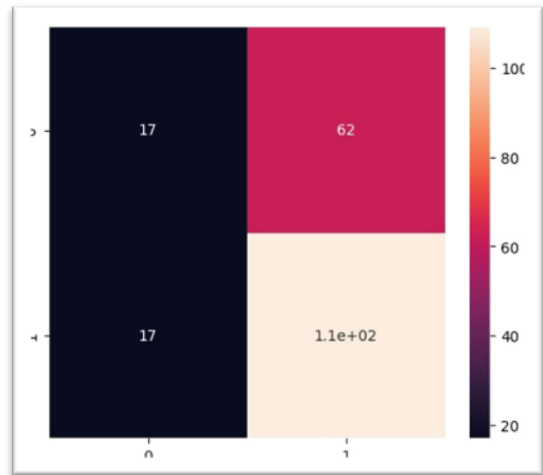


شکل ۸- نمودار Heat Map الگوریتم بیزین ساده

### ۴-۴- الگوریتم نزدیک‌ترین همسایه (KNN)

برای پیاده‌سازی الگوریتم KNN در پایتون از کتابخانه Scikit-Learn استفاده می‌شود که یکی از پرکاربردترین الگوریتم‌ها در حوزه 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.

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 الگوریتم SVM

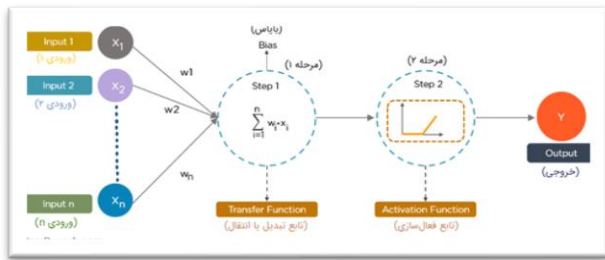
### ۴-۲- الگوریتم درخت تصمیم

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

“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 می‌شود، برای تعیین این‌که کدام نورون شلیک کند، اعمال می‌شوند. در این 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 می‌کند. 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. Keras می‌تواند 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 مدل‌های یادگیری عمیق است. مجموعه‌ای از کلاس‌ها و عملکردها را برای کمک به ایجاد و آموزش مدل‌های 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 مترکم 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.

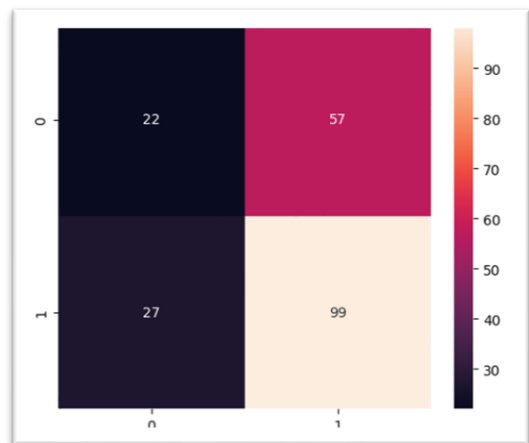
• **Flatten** از این کلاس برای صاف کردن لایه ورودی استفاده می‌شود.

• **Optimizers** این کلاس برای تعریف بهینه‌ساز مورد استفاده برای آموزش مدل استفاده می‌شود.

	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

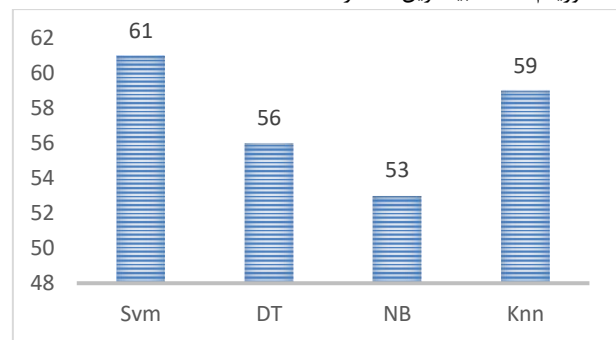
شکل ۹- خروجی الگوریتم KNN

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

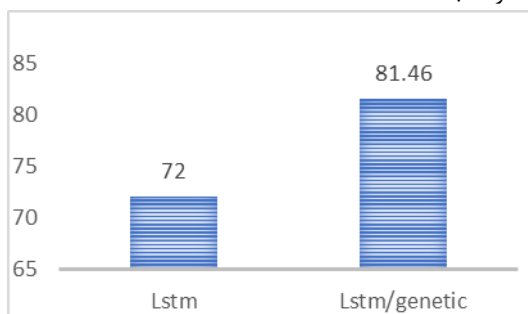
## ۵- مدل‌های شبکه عصبی

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 عصب‌ها 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

تعریف کنیم. در نهایت، باید عملگرهای ژنتیک مانند جهش، جابجایی و ترکیب را بر روی جمعیت اعمال کنیم تا نسل های جدیدی از شبکه های LSTM را تولید کنیم. به طور خلاصه، می توان گفت که الگوریتم ژنتیک بر LSTM یک روش بهینه سازی است که با شبیه سازی فرایند تکامل، می تواند بهترین شبکه LSTM را برای پیش بینی سری های زمانی چند متغیره انتخاب کند. این روش می تواند به شبکه LSTM کمک کند تا از اطلاعات گذشته بهتر استفاده کند و پیش بینی های دقیق تری ارائه دهد. انتخاب هایپر پارامتر مناسب یکی از مهم ترین مسائل در پیاده سازی مدل LSTM در مرحله ی آموزش است، دست یافتن به این هایپر پارامتر 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 می شود. بدین منظور ابتدا مقادیر مجاز برای هر پارامتر تعیین می شود. مقادیر مجاز برای این الگوریتم به شرح زیر است:

- Unit [۰.۸, ۱.۶, ۳.۲, ۶.۴]:
- Activation: ["sigmoid", "relu", "selu", "elu", "softmax", "softplus", "tanh"]
- Optimizer: ["adamax", "adam", "rmsprop", "nadam"]
- Loss: ["binary\_crossentropy", "mse", "mae"]
- Batch [۰.۲, ۰.۴, ۰.۸, ۱.۶, ۳.۲, ۶.۴]:

با استفاده از الگوریتم فرا ابتکاری ژنتیک هایپر پارامترهای تعداد واحد، تابع فعالساز، بهینه ساز، تابع ضرر و اندازه بسته ها بهینه می گردد. سپس با استفاده از این مقادیر مجاز یک جمعیت اولیه به تعداد ۱۰ کروموزوم ساخته می شود، هر کدام از ژن های یک کروموزوم شامل یک مقدار مجاز از هایپر پارامترهاست که انتخاب این مقادیر به صورت تصادفی است؛ بنابراین هر کدام از کروموزوم ها شامل ترکیبی بهینه از هایپر پارامتر می باشد که در آموزش مدل استفاده می شود. سپس ارزیابی کروموزوم ها با تابع برازش انجام می شود به این صورت که مدل LSTM به ازای هر کروموزوم یک بار اجرا می شود و بعد از آموزش با داده های آزمون ارزیابی می شود. بعد از ارزیابی جمعیت با استفاده از چرخ رولت کروموزوم های ارزنده بعنوان والد انتخاب می شوند. در مرحله بعدی کروموزوم های والد با روش چند نقطه ای باهم ترکیب شده و کروموزوم جدید که هایپر پارامترهای والدین را به ارث برده ایجاد می شود و در نهایت این کروموزوم بعد از جهش به جمعیت اضافه می شود. 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 نسل های الگوریتم که در این مقاله به تعداد ۵۰ نسل است به پایان نرسیده باشد، اجرای الگوریتم ادامه می یابد در غیر این صورت الگوریتم پایان می یابد و ژن های موجود در آن کروموزوم به عنوان هایپر پارامتر بهینه معرفی شده و در مرحله بعد ارزیابی می شوند و بهترین عضو آن ها به عنوان هایپر پارامتر ارزنده معرفی می شود و مدل LSTM را بهینه می کند. شکل ۱۴ نشان می دهد که صحت عملکرد این الگوریتم بر روی داده های 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

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

• Metrics از این کلاس برای تعریف معیارهای لازم استفاده می شود. در این قسمت مدل های شبکه عصبی LSTM و BERT که در این مقاله استفاده شده اند شرح داده می شوند. نظر به این که در تمام این مدل ها لایه تعبیه ساز یکسان است، پیاده سازی این لایه نیز توضیح داده می شود. این لایه در اولین لایه معماری مدل های شبکه عصبی عمیق قرار گرفته و بعنوان لایه ورودی در نظر گرفته می شود. برای پیاده سازی لایه تعبیه ساز در پایتون از لایه Embedding که در کتابخانه Keras تعریف شده است، استفاده می شود. پارامترهای این لایه که شامل تعداد لغات، اندازه بردار متراکم و طول نظرات که همان داده های ورودی هستند مقداردهی می شوند. مقدار ۳۰ برای اندازه بردار متراکم و طول نظرات در نظر گرفته شده است.

## ۱-۵- مدل LSTM

الگوریتم LSTM یک نوع شبکه عصبی بازگشتی است که می تواند از اطلاعات گذشته در پیش بینی آینده استفاده کند. این شبکه دارای یک حافظه داخلی است که می تواند اطلاعات مربوطه را نگه دارد و اطلاعات نامربوط را فراموش کند. پیاده سازی الگوریتم LSTM در پایتون نیاز به استفاده از کتابخانه هایی مانند تنسورفلو (Tensorflow) یا پایتورچ (PyTorch) دارد که امکان ساخت و آموزش شبکه های عصبی را فراهم می کنند. همچنین باید معماری و پارامترهای شبکه LSTM را با توجه به مسئله مورد نظر تعیین کرد. برای این کار، می توان از الگوریتم های فرا ابتکاری (Metaheuristic) مانند الگوریتم ژنتیک (Genetic Algorithm) که می تواند به بهینه سازی پارامترهای شبکه LSTM کمک کند استفاده کرد. این مدل به صورت ترتیبی ساخته می شود. 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 Dense با یک نورون برای تولید خروجی قرار می دهد. 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 می شود. همانطور که شکل ۱۳ نشان می دهد 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
weighted 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

## ۲-۵- بهینه سازی الگوریتم LSTM با ژنتیک

الگوریتم ژنتیک یک تکنیک بهینه سازی است که الهام گرفته از فرایند تکامل در طبیعت است. این الگوریتم با تولید و انتخاب جمعیت هایی از جواب های محتمل برای یک مسئله، سعی می کند به جواب بهینه برسد. پیاده سازی الگوریتم ژنتیک بر LSTM به این معنی است که می خواهیم معماری و پارامترهای شبکه LSTM را با استفاده از الگوریتم ژنتیک تنظیم کنیم. به این ترتیب، می توانیم بهترین شبکه LSTM را برای پیش بینی سری های زمانی چند متغیره پیدا کنیم. برای این کار، باید ابتدا جمعیت اولیه ای از شبکه های LSTM با معماری ها و پارامترهای مختلف تولید کنیم. سپس، باید تابع هدف و تابع برازندگی را برای ارزیابی عملکرد شبکه ها

استفاده می کند. با استفاده از این نمودارها می توان 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 می توان به اجرا روی چندین CPU، GPU و حتی سیستم عامل های تلفن های همراه اشاره کرد. نمودارها قابلیت ذخیره دارند و می توان آن ها را برای استفاده های بعدی ذخیره کرد. تمامی محاسبات درون نمودارها به تانسور ها وابسته است و به وسیله آن ها انجام می شود. 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 در جملات طراحی شده است. این الگوریتم از تکنیکی به نام ترانسفورمر استفاده می کند که به آن اجازه می دهد تا به صورت دو جهته متن را تحلیل کند. یعنی هم از سمت چپ به راست و هم از راست به چپ متن را بخواند و ارتباط بین کلمات را در نظر 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 می شود. این کتابخانه شامل ابزارهای مربوطه است. ابتدا مدل BERT PreTrained و Tokenizer که در سایت Huggingface قرار دارد و از قبل با داده های متنوع پیش آموزش دیده بارگزاری می شود و در ادامه با داده ها و اهداف پروژه تطبیق داده می شود. Huggingface یک پلتفرم مرکزی است که به ما امکان استفاده از جدیدترین و بهترین مدل ها و دیپاست های هوش مصنوعی را می دهد. سپس پارامترهای مدل مانند batch\_size, max\_lenght و تعیین می گردند، دسترسی به Gpu بررسی می گردد و در صورت وجود دیتاهای موجود در هر batch به Gpu انتقال می یابد و یک کلاس برای Tokenizer کردن داده ها ساخته می شود. این کلاس حداکثر طول جملات، Data و Tokenizer را می گیرد و دیتای مورد نیاز برای ورود به مدل BERT را می سازد. با استفاده از Tokenizer متن نظرات به بردارهای قابل فهم برای مدل تبدیل می شود. بردار شامل یکسری صفر و یک است که اهمیت / عدم اهمیت هر توکن را نشان می دهد. متغیر attention\_mask میان توکن های Pad و بامعنی تفاوت ایجاد می کند، صفر برای توکن های Pad و یک برای توکن های با معنی خواهد بود.

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

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

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

جدول ۳- بهترین عملکرد و مقادیر هایپر پارامترهای بهینه

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

### ۳-۵- مدل BERT

پیاده سازی الگوریتم BERT با زبان پایتون یکی از راه هایی است که می توان از این الگوریتم پیشرفته برای پردازش زبان طبیعی استفاده کرد. یکی از روش هایی است که می توانیم از این الگوریتم پیشرفته برای درک بهتر معنای کلمات و عبارات جستجو شده توسط کاربران استفاده کنیم. این الگوریتم از یک مدل ترانسفورمر دو طرفه استفاده می کند که می تواند با توجه به موقعیت و ارتباط کلمات در یک جمله، معنای آن ها را تشخیص دهد.

برای این کار نیاز به دانستن مفاهیم اساسی پایتون، تانسورفلو و ترانسفورمرز است. تانسورفلو یک کتابخانه یادگیری عمیق است که این امکان را فراهم می کند تا مدل های پیچیده را با استفاده از گراف های محاسباتی بسازیم. ترانسفورمرز یک کتابخانه است که شامل مجموعه ای از مدل های پردازش زبان طبیعی مبتنی بر BERT و مدل های مشابه است. با استفاده از این کتابخانه، می توانیم BERT را به راحتی بارگذاری کنیم، آموزش دهیم و ارزیابی کنیم.

کتابخانه تانسورفلو در پایتون؛ یک کتابخانه ریاضی نمادین است و با استفاده از داده ها و برنامه نویسی در فعالیت های مختلف شبکه های عصبی استفاده می شود. تانسورفلو مجموعه ای از مدل ها و الگوریتم های یادگیری ماشین و یادگیری عمیق است. تانسورفلو به ما این امکان را می دهد که نمودارها و ساختارهای جریان داده را بسازیم و با در نظر گرفتن ورودی ها به عنوان یک آرایه چند بعدی به نام تانسور، نحوه عملکرد آن ها را به صورت نمودار نشان دهیم. هم چنین می توانیم تمامی مراحل که داده ها طی می کنند را به صورت فلوجارت نمایش دهیم.

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

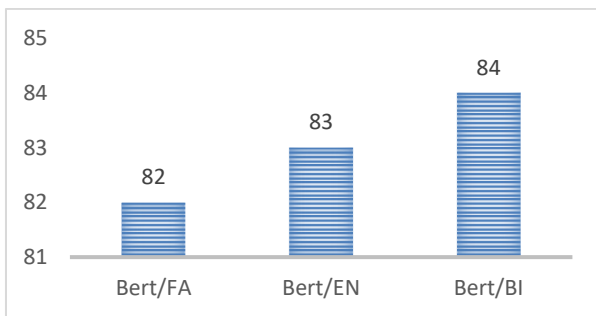
نام این کتابخانه از اصلی ترین قسمت اصلی آن یعنی Tensor گرفته شده است. تمامی محاسباتی که در تانسورفلو انجام می شود شامل تانسور است. اگر بخواهیم دقیق تر بگوییم که تانسور از چه چیزی تشکیل شده است، باید بگوییم که تانسور از یک ماتریس چند بعدی تشکیل شده است. این ماتریس می تواند هر نوع داده ای را نمایش دهد. البته این را هم باید بگوییم که درست است که تانسور قابلیت نمایش همه نوع داده را دارد اما تمامی داده های درون یک تانسور باید از یک نوع باشند. تانسور می تواند یک داده ورودی یا نتیجه یک محاسبه باشد. در کتابخانه تانسورفلو در پایتون تمامی عملیات روی نموداری که مجموعه ای از محاسبات متوالی است؛ انجام می شود. هر عملیاتی که انجام می شود را گره عملیات می گویند، تمامی گره ها به یکدیگر متصل هستند. نموداری که در طی عملیات تانسورفلو استفاده می شود؛ تمامی گره ها و اتصالات را نمایش می دهد. کتابخانه تانسورفلو در پایتون از نمودارها

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 کنیم. 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 الگوریتم BERT دو زبانه

شکل ۱۸ نشان 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 در این 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 الگوریتم‌های سنتی یادگیری ماشین و عملکرد آن‌ها بررسی شد. برای این منظور الگوریتم‌های درخت تصمیم، ماشین بردار پشتیبان، بیزین ساده و k نزدیک‌ترین همسایگی (KNN) پیاده سازی شدند. بردار پشتیبان با صحت ۶۱ درصد بهترین عملکرد را در میان الگوریتم‌های سنتی به دست آورد. در فاز بعدی به منظور مقایسه الگوریتم‌های یادگیری سنتی با الگوریتم‌های یادگیری عمیق مدل‌های LSTM و BERT به زبان فارسی و BERT به زبان انگلیسی طراحی شدند. این مدل‌ها به ترتیب با صحت‌های ۷۲ درصد، ۸۲ درصد و ۸۳ درصد عملکرد بهتری نسبت به الگوریتم‌های سنتی داشتند. در مرحله 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 ۹/۴۶ درصد نسبت به مدل اولیه LSTM پیشرفت داشت. در فاز پایانی این مقاله الگوریتم BERT دو زبانه با ترکیب متن فارسی نظرات و معنای

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 داده‌های آزمون اجرا شده و میزان خطا و دقت آموزش مدل محاسبه می‌گردد. 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 خروجی‌ها، مقدار خطا و دقت محاسبه می‌شود و نتایج چاپ می‌گردد.

پیاده سازی BERT به زبان انگلیسی هم دقیقاً مشابه BERT فارسی است با این تفاوت که از یک مدل BERT که روی داده‌های 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 هر کدام 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 می‌گردد.

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

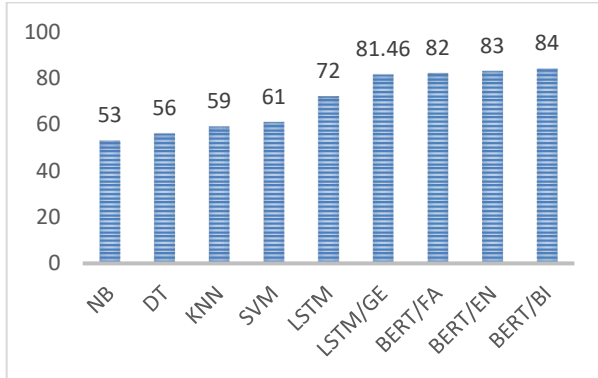
شکل ۱۶- نمایی از خروجی الگوریتم BERT به زبان انگلیسی

## 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،

مجموع	۸۳
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 می‌دهد.



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 الگوریتم‌های 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 الگوریتم‌های سنتی بدست آورد و مدل‌های یادگیر عمیق عملکرد بهتری نسبت به الگوریتم‌های سنتی داشتند. در ادامه مدل LSTM با استفاده از الگوریتم فرابتکاری ژنتیک با هدف به دست آوردن هابیر پارامترهای بهینه پیاده سازی شد. با کمک 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.

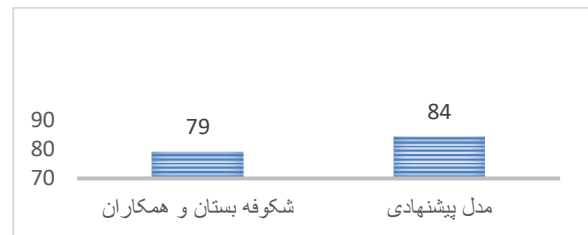
انتظار می‌رود این مدل عمیق چند زبانه نقش قابل توجهی در بهبود عملکرد سیستم‌های توصیه گر در این حوزه ایفا نماید. با توجه به اهمیت حجم دیتا در فرایند آموزش مدل‌های مبتنی بر شبکه‌های عصبی عمیق، می‌توان قبل از عملیات Fine-tune، مدل‌های زبانی را بر روی حجم زیادی از دیتای بدون برچسب در این حوزه آموزش داد که 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.

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 نسبتاً 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 دو کلاسه	۷۹ درصد	۸۴ درصد

همانطور که 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 هر کدام 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

## 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.