نشریــه علمــی دانش مالی تحلیل اوراق بهادار
دوره ۱۷ / شماره ۶۳ / پاییز ۱۴۰۳
صفحه ۷۷ تا ۹۲

طراحی سیستم معاملات الگوریتمی ترکیبی بهینهٔ یکپارچه با تخمین همزمان چندقیمتی

شیوا قاسم‌پور[۱]
شادی شاهوردیانی[*۲]
امیررضا کیقبادی[۳]
مهدی معدنچی زاج[۴]

تاریخ دریافت: ۱۴۰۲/۰۱/۳۰ تاریخ پذیرش: ۱۴۰۲/۰۵/۲۲

چکیده

طراحی سیستم‌های معاملاتی خودکار در کشورهای مختلف و اخیراً ایران افزایش یافته است. مزیت چنین سیستم‌هایی، افزایش سرعت و دقت تصمیم‌گیری معاملاتی بدون دخالت احساسات است. داشتن مدلی که بتواند به تخمین همزمان قیمت‌های باز شدن، بیشترین، کمترین و بسته شدن بپردازد، مزیت بسیار بزرگی برای سرمایه‌گذاران محسوب می‌شود. با ظهور فرایند یادگیری عمیق (*DNN*)، عصر جدید از شبکه‌های عصبی مصنوعی ایجاد شد، که توانایی‌های قابل توجهی در تشخیص الگو برای داده‌های زمانی طولانی دارند. در این تحقیق با استفاده از شبکه عصبی عمیق *LSTM* به تخمین همزمان چهار قیمت باز شدن، بیشترین، کمترین و بسته شدن پرداخته شده و به منظور دسته‌بندی نقاط خرید، فروش و نگهداشت از الگوریتم‌های مختلف دسته‌بندی استفاده و یک سیستم معاملات الگوریتمی یکپارچهٔ بهینه شده با الگوریتم فرا ابتکاری ژنتیک طراحی شده است. در فرایند دسته‌بندی، از روش یادگیری جمعی با استفاده از الگوریتم‌های درخت تصمیم، *KNN*، روش *RUSBoost* و شبکه عصبی پایه شعاعی (*RBN*) استفاده شده است. با توجه به اینکه برای مسائل و داده‌های زمانی مختلف، ویژگی‌های مختلفی وجود دارد و نمی‌توان برای مسائل دسته‌بندی یک سری ویژگی‌های کلی برای تمامی مسائل در نظر گرفت، لذا ویژگی‌های انتخابی بسته به نتایج و تفسیر محقق دارد. در این پژوهش، تعداد ۴۰ ویژگی قیمتی، حجمی و اندیکاتورهای حجمی و غیرحجمی در نظر گرفته شده است و ویژگی‌های بهینه از طریق فرایند بهینه‌سازی بصورت همزمان انتخاب و به منظور ارزیابی مدل پیشنهادی، از داده‌های سهام شپنا، خودرو و فجر بازار سرمایه ایران در بازه زمانی سال ۱۳۹۱ تا ۱۳۹۹ و نرم‌افزار *MATLAB* استفاده شده است. نتایج این پژوهش نشان داد که الگوریتم پیش‌بینی چندقیمتی توانسته است، هر چهار قیمت باز شدن، بالاترین، کمترین و بسته شدن را با دقت بالای ۹۰٪ تخمین بزند. همچنین مقادیر بهینه معیارهای عملکردی دسته‌بندی برای هر سه کلاس خرید، فروش و نگهداری از دقت بالایی برخوردار است و این نشان از اعتبار روش پیشنهادی و قوام آن است که قادر به تخمین و کلاسه‌بندی بهینه همزمان است. در نتیجه می‌توان روش پیشنهادی تحقیق حاضر را به عنوان یک روش بسیار مناسب برای معاملات خودکار در نظر گرفت و رباتی را بر اساس جهت انجام معاملات الگوریتمی در بازار سرمایه ایران طراحی و اجرایی کرد.

واژه‌های کلیدی: معاملات الگوریتمی، شبکه‌های عصبی عمیق، *LSTM*، بهینه‌سازی، تخمین چندقیمتی

[۱] دانشجوی دکتری مهندسی مالی، گروه مدیریت مالی، دانشکده مدیریت و اقتصاد، واحد علوم و تحقیقات، دانشگاه آزاد اسلامی، تهران، ایران. *shivaghasempour@gmail.com*

[۲] استادیار گروه مدیریت مالی، دانشکده علوم انسانی، واحد شهر قدس، دانشگاه آزاد اسلامی، تهران، ایران و استاد مدعو واحد علوم و تحقیقات، دانشگاه آزاد اسلامی، تهران، ایران. (نویسنده مسئول) *shshahverdiani@gmail.com*

[۳] استادیار، گروه حسابداری، دانشکده اقتصاد و حسابداری، واحد تهران مرکزی، دانشگاه آزاد اسلامی، تهران، ایران. a.keyghobadi@iauctb.ac.ir

[۴] استادیار، گروه مدیریت مالی، دانشکده مدیریت، واحد الکترونیکی، دانشگاه آزاد اسلامی، تهران، ایران. *madanchi@iauec.ac.ir*

۱- مقدمه

در حال حاضر، سیستم‌های معاملات الگوریتمی و خودکار به طور گسترده در بازارهای جهانی و بازار بورس اوراق بهادار تهران مورد استفاده قرار می‌گیرند. افزایش استفاده از این سیستم‌ها در ایران و کشورهای دیگر ناشی از تقاضای نهادهای مختلف برای طراحی و توسعه آن‌ها است. یکی از مزیت‌های اصلی این سیستم‌ها این است که به سرمایه‌گذاران امکان می‌دهند بدون تأثیر احساسات، تصمیمات معاملاتی را با سرعت و دقت بالا انجام دهند. برای طراحی و پیاده‌سازی این سیستم‌ها، از تکنیک‌های مختلفی براساس تحلیل تکنیکال، تحلیل بنیادی و تحلیل سری‌های زمانی استفاده شده است. این تکنیک‌ها را می‌توان به دو دسته آماری و هوش مصنوعی تقسیم کرد، که در میان تکنیک‌های مبتنی بر هوش مصنوعی، شبکه‌های عصبی از جمله روش‌های پراستفاده هستند (دینگ و کین[۱] ۲۰۲۰). شبکه‌های عصبی با استفاده از روند تغییرات داده‌های آموزش، یادگیری را انجام و سپس آن را به داده‌های تست تعمیم می‌دهند (مقدم و همکاران[۲]، ۲۰۱۶).

در سال‌های اخیر، با ظهور فرایند یادگیری عمیق[۳]، نسل جدیدی از شبکه‌های عصبی مصنوعی بوجود آمده است که از توانایی بسیار بالا در یادگیری و تشخیص الگو برای داده‌های زمانی طولانی را برخوردارند. وجود این ویژگی‌ها باعث افزایش تحقیقات در زمینه پیش‌بینی قیمت سهام با استفاده از شبکه‌های عصبی عمیق (*DNN*) شده است. در میان شبکه‌های عصبی عمیق، استفاده از شبکه‌های عصبی بازگشتی[۴] (*RNN*) برای پیش‌بینی قیمت سهام رایج‌تر بوده است. اما عیبی که *RNN* با آن مواجه است، آموزش دشوار آن بویژه در داده‌های دارای ساختار زمانی طولانی است. به منظور رفع این مشکل، شبکه‌های حافظه کوتاه ـ بلندمدت[۵] (*LSTM*) معرفی شده‌اند که با استفاده از مکانیزم گیت[۶]، اطلاعات خارجی را در لحظه فعلی در نظر می‌گیرند و توانایی یادگیری برای داده‌های زمانی طولانی را دارند. ساختار گیت‌دار *LSTM* توسط چوو[۷] و همکاران (۲۰۱۴) پیشنهاد شده است.

نکته قابل توجه این است که تمامی روش‌های فوق الذکر تاکنون تنها برای پیش‌بینی تک قیمتی (معمولاً قیمت بسته شدن) استفاده شده‌اند. برای پیش‌بینی چندین متغیر به صورت همزمان، نیاز به طراحی مدلی است که با چندین ورودی و چندین خروجی کار کند. در این تحقیق، مدل *LSTM-DRNN* براساس مدل دینگ و کین طراحی و اجرا شده است که توانایی تخمین همزمان چهار قیمت باز شدن، بیشترین، کمترین و بسته شدن را دارد. علاوه بر این، بسیاری از سیستم‌های معاملات الگوریتمی پیشنهادی توسط محققین بر اساس یک یا چند روش مستقل و یا ترکیبی از الگوریتم‌ها طراحی شده‌اند. در این تحقیق، با استفاده از رویکرد یادگیری جمعی[۸] و جداسازی بخش تخمین قیمت از بخش دسته‌بندی[۹]، یک رویکرد ترکیبی یکپارچه پیشنهاد شده است. این رویکرد بهینه‌سازی همزمان انتخاب ویژگی‌های بهینه و پارامترهای الگوریتم‌های دسته‌بندی و رگرسیون را فراهم می‌کند. برای بهینه‌سازی، از الگوریتم فراابتکاری ژنتیک استفاده می‌شود.

۲- مبانی نظری و مروری بر پیشینه پژوهش

۲-۱- معاملات الگوریتمی

در بازارهای مالی الکترونیکی، معاملات الگوریتمی به معنای استفاده از برنامه‌های کامپیوتری برای ورود سفارش‌های معاملاتی است که سیگنال‌های معاملاتی آن

---

[1] *Ding & Qin*
[2] *Moghaddam*
[3] *Deep learning*
[4] *Recurrent neural network*
[5] *Long-short term memory*
[6] *Gate*
[7] *Cho*
[8] *Ensemble learning*
[9] *Classification*

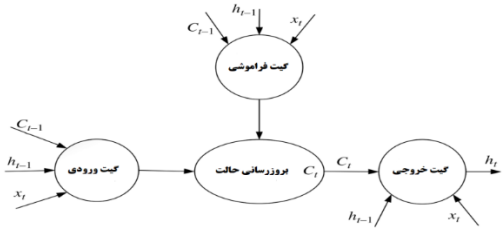
توسط بخشی دیگر از سیستم تولید شده است (سیفرت[۱]، ۲۰۱۸). الگوریتم‌های معاملاتی را می‌توان به سه دسته الگوریتم‌های اجرا[۲]، معاملات کسب سود[۳] و الگوریتم‌های معاملاتی پربسامد[۴] تقسیم کرد (کیسل[۵]، ۲۰۲۱):

- الگوریتم‌های اجرا: الگوریتم اجرا وظیفه اجرای تصمیم سرمایه‌گذاری اتخاذشده توسط سرمایه‌گذاران یا مدیریت سبد سرمایه‌گذاری را بر عهده دارد.
- الگوریتم‌های جستجوی سود: الگوریتمی است که هم خرید و یا فروش در بازار را تعیین می‌کند و هم آن تصمیمات را بدون تعامل با سرمایه‌گذار انجام می‌دهد.
- معاملات پربسامد: معاملات پربسامد (HFT) نوعی از الگوریتم جستجوی سود است که به جستجوی سود معاملاتی کوتاه‌مدت است.

۲-۲- تخمین قیمت

شبکه‌های عصبی، به ویژه شبکه‌های عصبی بازگشتی (RNN)، در پیش‌بینی قیمت سهام در بازار سرمایه کاربرد گسترده‌ای دارند. شبکه‌های عصبی بازگشتی به دلیل قابلیت یادگیری عمیق، توانایی بالا در پیش‌بینی با دقت بالا را دارا می‌باشند. با این حال، یکی از مشکلات شبکه‌های عصبی بازگشتی این است که در یادگیری وابستگی‌های طولانی مدت ضعف دارند. به منظور رفع این مشکل، مفهوم حافظه کوتاه ـ بلندمدت ($LSTM$) معرفی شده است. در شبکه $LSTM$، نرون‌ها قادر به ذخیره‌سازی چهار حالت مقدار فعلی و آخرین خروجی، مقدار فعلی و آخرین مقدار از حالت نرون حافظه هستند. در واقع، شبکه $LSTM$ یک نوع خاص از شبکه عصبی بازگشتی است که ساختاری با سه گیت دارد.

شبکه $LSTM$ با استفاده از یک لایه شبکه عصبی سیگموید (معادله (۱)) و یک جفت عملیات ضربی، تصمیم می‌گیرد که آیا اطلاعات عبور داده شوند یا خیر. هر عنصر خروجی از لایه سیگموید یک عدد حقیقی بین [۰،۱] است که نمایانگر وزن اطلاعات عبوری است.



شکل ۱- ساختار شبکه *LSTM*.

علاوه بر این، به منظور بروزرسانی نورون‌ها، از یک لایه شامل تابع فعالسازی $tanh$ مطابق معادله (۲) استفاده می‌شود (دینگ و کین، ۲۰۲۰):

$$\sigma(x)=\frac{1}{1+e^{-x}} \qquad (۱)$$

$$\tanh(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \qquad (۲)$$

گیت فراموشی که به منظور تعیین اطلاعات رد شده بکار می‌رود، اطلاعات h_{t-1} و x_t را گرفته و حالت نورون C_t با مقداری بین ۱-۰ را برمی‌گرداند. معادله (۳) روند محاسبه احتمال فراموشی را نشان می‌دهد:

$$f_t=\sigma\left(W_f\cdot\left[h_{t-1},x_t\right]+b_f\right) \qquad (۳)$$

که در آن h_{t-1} بیانگر خروجی نورون قبلی و x_t ورودی نورون فعلی است.

گیت ورودی تعیین می‌کند که چه مقدار اطلاعات جدید به حالت نورون اضافه شود. ابتدا، لایه ورودی که شامل تابع فعالسازی سیگموید است، تعیین می‌کند که کدام یک از اطلاعات نیاز به بروزرسانی دارند، و سپس لایه $tanh$ بردارهای کاندید $\hat{C}_t$ را تولید کرده و حالت نورون‌ها مطابق با معادله (۴) بروزرسانی می‌شود:

$$C_t=f_t\times C_{t-1}+i_t\times\hat{C}_t \qquad (۴)$$

که در آن i_t و $\hat{C}_t$ از روابط زیر بدست می‌آیند:

$$i_t=\sigma\left(W_i\cdot\left[h_{t-1},x_t\right]+b_i\right) \qquad (۵)$$

$$\hat{C}_t=\tanh\left(W_c\cdot\left[h_{t-1},x_t\right]+b_c\right) \qquad (۶)$$

گیت خروجی در شبکه $LSTM$ به منظور کنترل فیلتر کردن حالت فعلی نورون و حالت‌های کنترلی استفاده می‌شود:

---

[۱] *Seyfert*

[۲] *Execution algorithm*

[۳] *Profit seeking algorithm*

[۴] *High frequency algorithm*

[۵] *Kissell*

$$O_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \quad (۷)$$

$$h_t = O_t \times \tanh\left(C_t\right) \quad (۸)$$

۲-۳- پیشینه تحقیق

بطور کلی، تحقیقات در زمینه معاملات الگوریتمی می‌توانند به چند دسته کلی تقسیم شوند:

۱. روش‌های مبتنی بر تحلیل تکنیکال: این دسته شامل تحقیقاتی است که بر تحلیل نمودارهای قیمت و حجم سابقه‌ی سهام تمرکز دارند. روش‌هایی مانند استفاده از مدل‌های مبتنی بر الگوهای نموداری، شاخص‌های تحلیلی و روش‌های پیشرفته‌تر مانند شبکه‌های عصبی در این دسته مطرح هستند. در بین این روش‌ها می‌توان به تحقیق سزار[۱] و همکاران(۲۰۱۷)، هروانی و خلیلی‌عراقی(۱۳۹۹)، اسماعلیل و همکاران(۱۴۰۰)، وو و یاست ـ برم[۲](۲۰۲۰)، کوهن[۳](۲۰۲۳)، گرمانوف[۴](۲۰۲۳) اشاره کرد.

۲. روش‌های مبتنی بر تحلیل بنیادی: این دسته شامل تحقیقاتی است که بر اساس اطلاعات و اخبار مالی و اقتصادی تمرکز دارند. استفاده از شاخص‌های مالی، اخبار شرکت‌ها، گزارش‌های مالی و موارد مشابه برای تصمیم‌گیری در معاملات این دسته از تحقیقات استفاده می‌کنند. تحقیق ایامکانیچت[۵] و همکاران(۲۰۱۷) و هوانگ[۶] و همکاران(۲۰۱۹) در زمره این دسته قرار دارند.

۳. روش‌های مبتنی بر یادگیری ماشین در زمینه معاملات الگوریتمی، از الگوریتم‌ها و تکنیک‌های یادگیری ماشین برای تحلیل داده‌ها و پیش‌بینی رفتار سهام استفاده می‌کنند. این روش‌ها با استفاده از الگوریتم‌های یادگیری ماشین قادرند الگوها و روندهای مهم را از داده‌های مالی استخراج کرده و بهبود دقت پیش‌بینی و تصمیم‌گیری معاملاتی را ارائه دهند. به طور کلی، روش‌های مبتنی بر یادگیری ماشین در معاملات الگوریتمی عملکردی مشابه به عملکرد بشر دارند. آنها از داده‌های تاریخی مرتبط با سهام و اطلاعات مربوط به بازار استفاده می‌کنند و با استفاده از الگوریتم‌های یادگیری ماشین، الگوها، ارتباطات و قوانین مهم را در داده‌ها شناسایی می‌کنند. مهمترین روش‌های مبتنی بر یادگیری ماشین در معاملات الگوریتمی عبارتند از:

- روش‌های یادگیری تقویتی[۷] (RL): روش‌های یادگیری تقویتی یک رویکرد قدرتمند در معاملات الگوریتمی هستند که بر اساس تعامل عامل (معامله‌گر) با محیط و دریافت برخوردهای مثبت یا منفی برای تصمیم‌های خود، بهبود مداوم عملکرد خود را هدف می‌گیرند. این روش‌ها تلاش می‌کنند با انجام عملیات تجربه و خطا، یادگیری عملکرد بهینه را در معاملات تقویت کنند. در روش‌های یادگیری تقویتی، یک عامل (معامله‌گر) در یک محیط معاملاتی عمل می‌کند. عامل با دریافت وضعیت فعلی محیط و با استفاده از سیاست‌های تصمیم‌گیری، تصمیم‌های عملیاتی را انجام می‌دهد. سپس عامل پاداش یا مجازاتی بر اساس عملکرد خود دریافت می‌کند که به عنوان بازخورد به عملکرد خود عمل می‌کند. چهرقانی و رستگار(۱۳۹۸)، پانامارف[۸] و همکاران(۲۰۱۹)، لی و همکاران[۹](۲۰۲۰)، شاوندنی و خدمتی[۱۰](۲۰۲۲)، فیلزاردو[۱۱] و همکاران(۲۰۲۲) و لیو و همکاران(۲۰۲۳) از روش‌های یادگیری تقویتی و تقویتی عمیق برای معاملات الگوریتمی استفاده کرده‌اند.
- شبکه‌های عصبی عمیق (DNN): این روش‌ها از ساختار چند لایه و متصل به همین عنوان استفاده می‌کنند. آنها قادرند الگوهای پیچیده و مفهوم‌های مهم را از داده‌های مالی استخراج کرده و بهبود دقت پیش‌بینی را ارائه دهند. این شبکه‌ها قابلیت یادگیری توابع پیچیده را دارند و می‌توانند از الگوها و ویژگی‌های پنهان در داده‌های معاملاتی بهره ببرند. یافتیان و رستگار(۱۳۹۷) و تیونینگا(۲۰۲۳) از یادگیری عمیق مبتنی بر شبکه عصبی پیچشی[۱۲] (CNN) برای

---

۱ *Sezer*

۲ *Vo & Yost-Bremm*

۳ *Cohen*

۴ *Germanov*

۵ *Eiamkanitchat*

۶ *Huang*

۷ *Reinforcement learning*

۸ *Ponomarev*

۹ *Lei*

۱۰ *Shavandi & Khedmati*

۱۱ *Felizardo*

۱۲ *Convolutional neural network*

معاملات الگوریتمی بهره گرفته‌اند.همچنین هوانگ[۱] و همکاران(۲۰۲۳) از یادگیری Q عمیق برای معاملات الگوریتم استفاده کرده‌اند.

- شبکه‌های عصبی بازگشتی (RNN): این روش‌ها به خاطر قدرتشان در کشف الگوهای زمانی و وابستگی‌های طولانی مدت مورد توجه قرار گرفته‌اند. آنها می‌توانند تاریخچه‌ی زمانی را در نظر بگیرند و رفتار سهام را در طول زمان پیش‌بینی کنند. کائو[۲] و همکاران(۲۰۲۳) و صیقلی[۳] و همکاران(۲۰۲۳) از شبکه‌های RNN برای طراحی معاملات الگوریتمی استفاده کرده‌اند.
- شبکه‌های عصبی $LSTM$: این شبکه‌ها نوع خاصی از شبکه‌های RNN هستند که به خاطر قابلیت ذخیره‌سازی و حفظ اطلاعات برای مدت زمان طولانی در طول توالی‌های زمانی، در معاملات الگوریتمی استفاده می‌شوند. شبکه $LSTM$ به دلیل قدرت خود در تشخیص الگوها و وابستگی‌های طولانی مدت، به عنوان یک ابزار قدرتمند در معاملات الگوریتمی مورد استفاده قرار می‌گیرد. این روش قادر است به طور همزمان، اطلاعات زمانی و سابقه قیمتی را در نظر بگیرد و پیش‌بینی دقیقی از رفتار سهام ارائه دهد. همچنین، با استفاده از روش‌های بهینه‌سازی و پیشرفته، عملکرد شبکه $LSTM$ در معاملات الگوریتمی قابل بهبود است. ون و یوآن[۴](۲۰۱۸) از ترکیب شبکه $LSTM$ و CNN، روندو[۵](۲۰۱۹) از شبکه $LSTM$، دینگ و کین(۲۰۲۰) از شبکه $DRNN\text{-}LSTM$ و ناگاراج و همکاران(۲۰۲۳) از شبکه $LSTM\text{-}ANN$ برای طراحی معاملات الگوریتم بهره برده‌اند.

این روش‌ها می‌توانند به صورت مستقل یا ترکیبی استفاده شوند و با توجه به ماهیت داده‌ها و مسئله معاملاتی، روش مناسبی انتخاب شود. همچنین، می‌توان با بهینه‌سازی پارامترها و استفاده از الگوریتم‌های پیشرفته، عملکرد این روش‌ها را بهبود داد.

۴. روش‌های مبتنی بر بهینه‌سازی: این دسته شامل تحقیقاتی است که از الگوریتم‌های بهینه‌سازی برای پیدا کردن بهترین استراتژی معاملاتی و بهینه‌سازی پارامترهای مدل استفاده می‌کنند. الگوریتم‌هایی مانند الگوریتم ژنتیک، جستجوی همزمان و مولفه‌های دیگر به منظور بهبود عملکرد مدل‌های معاملات الگوریتمی در این دسته ارائه می‌شوند. به عنوان مثال، کانتی و لوپز[۶](۲۰۱۹) و کریستودولاکی[۷] و همکاران(۲۰۲۳) از الگوریتم ژنتیک، مزومدار[۸] و همکاران(۲۰۱۹) از الگوریتم گرگ خاکستری و خاندلوال و تک[۹](۲۰۲۲) از الگوریتم ازدحام ذرات برای بهینه‌سازی سیستم معاملات الگوریتمی خود بهره برده‌اند.

همچنین، تحقیقات معاملات الگوریتمی به طور معمول از ترکیبی از این دسته‌بندی‌ها و روش‌ها برای بهبود عملکرد و دقت استفاده می‌کنند. به عنوان مثال، موسوی انزهایی و نیکومرام(۱۳۹۹) با استفاده از مدلی ترکیبی از آینده‌پژوهشی، تحلیل بنیادی، قواعد معاملاتی خبرگان و الگوریتم‌های یادگیری ماشین به طراحی معاملات الگوریتم پرداخته است. همچنین آیسل و سانتور[۱۰](۲۰۲۲) از ترکیب تحلیل تکنیکال و یادگیری جمعی به طراحی یک سیستم معاملات الگوریتم پرداخته‌اند.

۳- روش تحقیق

روش تحقیق حاضر دارای ۵ مرحله زیر است:

۱- برچسب‌گذاری داده‌های سری زمانی
۲- نرمال‌سازی داده‌ها
۳- دسته‌بندی و انتخاب ویژگی
۴- تخمین قیمت
۵- بهینه‌سازی

---

۱ *Huang*
۲ *Cao*
۳ *Seighaly*
4 *Wen & Yuan*
5 *Rundo*
۶ *Conti & Lopes*
۷ *Christodoulaki*
۸ *Mazumdar*
۹ *Khandelwal & Tak*
۱۰ *AYCEL & SANTUR*

تمامی مراحل برچسب‌گذاری، نرمال‌سازی، دسته‌بندی و انتخاب ویژگی، تخمین قیمت و بهینه‌سازی مکمل یکدیگر هستند و فرایند بهینه‌سازی بصورت یک فرایند یکپارچه صورت می‌گیرد.

۳-۱- برچسب‌گذاری داده‌های سری زمانی

داده‌های مورد استفاده در این تحقیق، معاملات روزانه برخی از سهام منتخب بازار اوراق بهادار تهران است. این داده‌ها از تاریخ فروردین سال ۱۳۹۱ تا انتهای اسفند سال ۱۳۹۹ جمع‌آوری شده است. متغیرهای ورودی مورد بررسی در تحقیق حاضر عبارتند از داده‌های قیمتی (باز شدن (*Open*)، بیشترین (*High*)، کمترین (*Low*) و بسته شدن (*Close*)) به همراه حجم معاملات و اندیکاتور[۱]های غیرحجمی و حجمی. این متغیرها به عنوان ویژگی برای فرایند دسته‌بندی و تخمین قیمت استفاده گردیده است. اندیکاتورهای غیرحجمی وابسته به داده‌های قیمتی و اندیکارتوهای حجمی وابسته به داده‌های حجم و (یا) قیمت معاملات هستند.

در این تحقیق اسیلاتور[۲]ها، استوکستیک[۳]ها، اندیس[۴]ها و اندیکاتورها را بصورت خلاصه اندیکاتور می‌نامیم. اندیکاتورهای غیرحجمی در جدول ۱ و اندیکاتورهای حجمی در جدول ۲ نشان داده شده است. بطور کلی با در نظر گرفتن قیمت‌های باز شدن، بیشترین، کمترین، بسته شدن و حجم معاملات، همراه با ۲۹ اندیکاتور غیرحجمی و ۶ اندیکاتور حجمی، در مجموع تعداد ۴۰ ویژگی در این تحقیق در نظر گرفته شده است.

برای انجام برچسب‌گذاری روی داده‌های سری زمانی در معاملات الگوریتمی، از سری زمانی قیمت بسته شدن استفاده شده است. در این تحقیق برچسب‌گذاری نقاط خرید، فروش و نگهداری با استفاده از روش پنجره لغزشی[۵] انجام شده است. در تکنیک پنجره لغزشی با توجه به طول پنجره، در هر لغزش یک گام (در اینجا با توجه به اینکه داده‌های مورد استفاده بصورت روزانه است، یک روز) به جلوتر می‌رود و این عمل تا رسیدن به پنجره آخر ادامه پیدا می‌کند.

هزینه تراکنش در بازار بورس اوراق بهادار تهران یا درواقع کارمزد کارگزاری برای خرید برابر ۰٫۳۷۱۲ درصد و برای فروش برابر ۰٫۸۸ درصد و در مجموع برای یک معامله خرید و فروش برابر با ۱٫۲۵۱۲ درصد است. با توجه به اینکه هدف ما از خرید و فروش یک سهم کسب سود است، بنابراین حداقل سود ۱ درصد را به عنوان کمترین میزان سود قابل قبول برای معاملات الگوریتمی در نظر گرفته شده است. بنابراین در فرایند برچسب‌گذاری قیدی برابر با ۲٫۲۵۱۲ درصد در نظر گرفته شد، چنانچه نقطه خرید و فروش (انجام یک معامله) دارای سودی بیشتر از این آستانه بود، برچسب‌گذاری شد و در غیر اینصورت وارد پنجره بعدی شدیم تا زمانی‌که نقطه فروش مناسب پیدا شود.

جدول ۱- اندیکاتورهای غیرحجمی مورد مطالعه

ردیف	اندیکاتور	توضیح	داده‌های مورد نیاز	دوره زمانی
۱	*A/D.O*	*Accumulation/Distribution Oscillator*	*Open, High, Low, Close*	ندارد
۲	*S.O*	*Stochastic oscillator*	*High, Low, Close*	دارد
۳	*Acceleration*	*Acceleration Between Times*	*Open, High, Low, Close*	دارد
۴	*MOM*	*Momentum Between Times*	*Open, High, Low, Close*	دارد
۵	*SMA*	*Simple Moving Average*	*Close*	دارد
۶	*EMA*	*Exponential Moving Average*	*Close*	دارد
۷	*MMA*	*Modified Moving Average*	*Close*	دارد
۸	*MACD*	*Moving Average Convergence/Divergence*	*Open, High, Low, Close*	ندارد
۹	*RSI*	*Relative Strength Index*	*Close*	دارد

[۱] *Indicator*
[۲] *Oscillator*
[۳] *Stochastic*
[۴] *Index*
[۵] *Sliding window*

۱۰	*Williams %R*	*Williams %R*	*High, Low, Close*	دارد
۱۱	*Bollinger*	*Middle, Upper, and Lower Bands*	*Close*	دارد
۱۲	*ROC*	*Price Rate of Change*	*Close*	دارد
۱۳	*WADLine*	*Williams Accumulation/Distribution line*	*High, Low, Close*	ندارد
۱۴	*Typical Price*	*Average of the high, low, and closing prices*	*High, Low, Close*	ندارد
۱۵	*Chaikin Volatility*	*Chaikin Volatility*	*High, Low*	دارد

جدول ۲- اندیکاتورهای حجمی مورد مطالعه

ردیف	اندیکاتور	توضیح	داده‌های مورد نیاز	دوره زمانی
۱	*NVI*	*Negative Volume Index*	*Close, Volume*	ندارد
۲	*PVI*	*Positive Volume Index*	*Close, Volume*	ندارد
۳	*OBV*	*On-Balance Volume*	*Close, Volume*	ندارد
۴	*PVT*	*Price and Volume Trend*	*Volume*	ندارد
۵	*VROC*	*Volume Rate of Change*	*Volume*	دارد

با توجه به اهمیت بالای انتخاب طول پنجره مناسب، به منظور انتخاب بهترین طول پنجره از روش آنالیز حساسیت استفاده گردید. در این روش، حساسیت طول پنجره و تعداد نقاط خرید یا فروش بررسی و طول پنجره‌ای که تعداد نقاط خرید و فروش بیشتری را به دنبال داشت، به عنوان طول پنجره مناسب انتخاب شد. با انجام آنالیز حساسیت، این تحقیق به تعیین بهترین پارامتر برای طول پنجره در معاملات الگوریتمی می‌پردازد.

۳-۲- نرمال‌سازی داده‌ها

یکی از روش‌های پرکاربرد برای نرمال‌سازی داده‌ها، روش *min-max* است که داده‌ها را در محدوده بین ۰ تا ۱ قرار می‌دهد.

۳-۳- دسته‌بندی و انتخاب ویژگی بهینه

به منظور انجام عمل دسته‌بندی و انتخاب ویژگی‌ها بر اساس معیارهای سنجش عملکرد الگوریتم‌های دسته‌بندی، ابتدا داده‌ها با روش اعتبارسنجی مقطعی[۱] *k-fold* به دو دسته آموزش و تست تقسیم شد. در این روش، داده‌ها به k بخش تقسیم شده و به ازای k بار، بخش‌های مختلف به عنوان داده‌های آموزش و تست انتخاب می‌شوند و سپس الگوریتم‌های دسته‌بندی بر روی داده‌های آموزش اجرا و عملکرد آنها با استفاده از داده‌های تست محاسبه می‌شود. نتایج عملکرد الگوریتم‌ها برای هر بار تکرار به دست آمده و در نهایت میانگین این نتایج برای ارزیابی نهایی استفاده می‌شود. در این تحقیق، از روش ۵-*fold* استفاده گردید که در آن، داده‌ها به ۵ بخش تقسیم شده و به طور متوسط ۸۰٪ از داده‌ها به عنوان داده‌های آموزش و ۲۰٪ مابقی به عنوان داده‌های تست استفاده شد.

با توجه به اینکه در معاملات الگوریتم با سه کلاس خرید، فروش و نگهداری سروکار داریم، بنابراین باید از روش‌های دسته‌بندی چندکلاسه استفاده شود. روش‌های مختلفی برای دسته‌بندی چندکلاسه وجود دارد که در این مطالعه از ۴ الگوریتم درخت تصمیم (*DTree*)، *K* - نزدیکترین همسایگی (*KNN*)، روش *RUSBoost* و شبکه عصبی پایه شعاعی[۲] (*RBN*) استفاده شده است. پس از تست این الگوریتم‌ها با داده‌های تست، وزن بدست آمده از فرایند بهینه‌سازی برای هر کدام از الگوریتم‌ها در نظر گرفته شده و میانگین وزن‌دار معیارهای عملکردی دسته‌بندی محاسبه گردید. غالباً در معاملات الگوریتمی، ۴ معیار عملکردی زیر در نظر گرفته می‌شود:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \quad (۱۰)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \quad (۱۱)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \quad (۱۲)$$

[۱] *Cross-validation*

[۲] *Radial basis network*

$$F1\text{-}Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (۱۳)$$

که در آن TP[۱] بیانگر کلاس مثبتی است که بدرستی مثبت تشخیص داده شده است. TN کلاس منفی هست که بدرستی منفی تشخیص داده شده است. FP بیانگر کلاس منفی است که مثبت تشخیص داده شده است و FN نیز بیانگر کلاس مثبت است که به اشتباه منفی تشخیص داده شده است. مقادیر هر یک از معیارهای چهارگانه بین ۰ و ۱ قرار دارد، و هرچقدر این مقدار به ۱ نزدیک باشد، بیانگر عملکرد خوب الگوریتم‌های دسته‌بندی است.

با در نظر گرفتن اینکه در معاملات الگوریتم با سه کلاس روبرو هستیم، بنابراین چهار معیار فوق بطور جداگانه برای هر کلاس محاسبه و در نهایت مقدار میانگین وزن‌دار آنها به عنوان مقدار نهایی بدست آمد. روند اینکار در جدول ۳ نشان داده شده است.

در نهایت نیز در انتهای این مرحله، بر اساس فرایند بهینه‌سازی و تابع برازندگی در نظر گرفته شده، ویژگی‌های بهینه از بین ۴۰ ویژگی مورد مطالعه انتخاب گردید. روند بهینه‌سازی در بخش مربوطه تشریح شده است. ویژگی‌ها بصورت باینری (۱ و ۰) در نظر گرفته شدند و آن دسته از ویژگی‌هایی که مقدار بهینه آنها برابر ۱ بود، از طریق فرایند بهینه‌سازی انتخاب گردید.

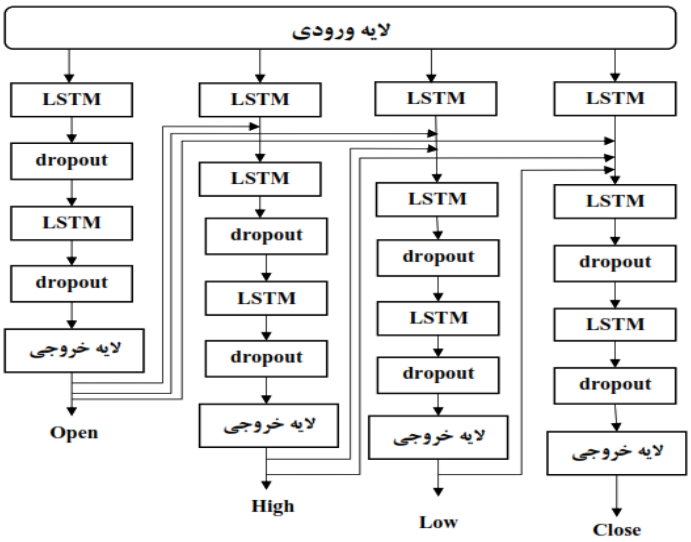
۳-۴- تخمین قیمت

در این تحقیق با الگو گرفتن از تحقیق دینگ و کین(۲۰۲۰)، شبکه *LSTM-DRNN* طراحی شد که قادر است همزمان هر چهار قیمت باز شدن، بیشتری و کمترین و قیمت بسته شدن را پیش‌بینی نماید. در *DRNN* به منظور جلوگیری از بیش‌برازش شبکه، از روش *dropout* (به معنای رها کردن) استفاده شده است. مدل شبکه عصبی بازگشتی مبتنی بر *LSTM* با لایه *dropout* تحت عنوان *LSTM-DRNN* در شکل ۲ نشان داده شده است که قادر است بصورت همزمان چهار قیمت باز شدن، بیشترین، کمترین و بسته شدن را تخمین بزند.

جدول ۳- روند محاسبه مقادیر میانگین وزن‌دار معیارهای عملکردی

	الگوریتم ۱ (*DTree*)	الگوریتم ۲ (*KNN*)	الگوریتم ۳ (*RUSBoost*)	الگوریتم ۴ (*ERBN*)	مقدار میانگین
$\overline{\text{Accuracy}}$	W_1	W_2	W_3	W_4	$\sum_{i=1}^{4} W_i \ \overline{\text{Accuracy}}_i$
$\overline{\text{Precision}}$	W_1	W_2	W_3	W_4	$\sum_{i=1}^{4} W_i \ \overline{\text{Precision}}_i$
$\overline{\text{Recall}}$	W_1	W_2	W_3	W_4	$\sum_{i=1}^{4} W_i \ \overline{\text{Recall}}_i$
$\overline{F1\text{-}Score}$	W_1	W_2	W_3	W_4	$\sum_{i=1}^{4} W_i \ \overline{F1\text{-}Score}_i$

[۱] *True-Positive*



شکل ۲- ساختار شبکه *LSTM-DRNN* پیشنهادی.

جدول ۴- روند محاسبه مقادیر میانگین وزن‌دار قیمت‌ها

	الگوریتم ۱ (*LSTM-DRNN*)	الگوریتم ۲ (*GRNN-DRNN*)	الگوریتم ۳ (*LSBoost-DRNN*)	مقدار میانگین
Open	W_1	W_2	W_3	$\sum_{i=1}^{3} W_i \mathrm{Open}_i$
High	W_1	W_2	W_3	$\sum_{i=1}^{3} W_i \mathrm{High}_i$
Low	W_1	W_2	W_3	$\sum_{i=1}^{3} W_i \mathrm{Low}_i$
Close	W_1	W_2	W_3	$\sum_{i=1}^{3} W_i \mathrm{Close}_i$

برای انجام تخمین قیمت، علاوه بر مدل *LSTM-DRNN*، از دو مدل شبکه عصبی رگرسیون تعمیم‌یافته[1] (*GRNN*) و روش رگرسیون مبتنی بر یادگیری جمعی بوستینگ حداقل مربعات[2] (*LSBoost*) استفاده شده است. در نهایت مقدار تخمینی هر یک از قیمت‌ها برابر با مقدار میانگین وزن‌دار این الگوریتم‌ها است. وزن هر یک از الگوریتم‌ها نیز از طریق بهینه‌سازی بدست آمد. روند این تخمین مطابق با جدول ۴ است.

۳-۵- فرایند بهینه‌سازی

در این تحقیق، فرایند بهینه‌سازی برای مراحل دسته‌بندی و انتخاب ویژگی و همچنین مرحله تخمین قیمت به صورت یکپارچه و همزمان انجام گردید. در هر دو مرحله، الگوریتم‌ها دارای یک سری پارامتر به عنوان تنظیمات الگوریتم بودند و مقادیر این پارامترها از طریق فرایند بهینه‌سازی بدست آمدند. همچنین، برای انتخاب ویژگی بهینه نیز از تکنیک بهینه‌سازی استفاده شد و ویژگی‌ها به عنوان متغیرهای تصمیم در نظر گرفته شدند. در جدول ۴، پارامترهای بهینه‌سازی همراه با کران بالا و کران پایین آنها آورده شده است. برای متغیرهای مربوط به ویژگی‌ها، کران پایین برابر ۰ و کران بالا برابر ۱ است، که به این صورت به متغیرهای باینری تبدیل شدند. همچنین، برای متغیرهای مربوط به الگوریتم‌های دسته‌بندی و رگرسیون و متغیرهای وزن، کرانهای خاصی تعیین شده است. با توجه به جدول ۴، در کل ۵۵ متغیر تصمیم از فرایند بهینه‌سازی بدست آمده است. از این ۵۵ متغیر، ۱۱ متغیر تصمیم مربوط به الگوریتم‌های دسته‌بندی و رگرسیون، ۴۰ متغیر تصمیم مربوط به ویژگی‌ها و ۷ متغیر تصمیم مربوط به وزن هر یک از الگوریتم‌های دسته‌بندی و رگرسیون می باشد. این متغیرها در فرایند بهینه‌سازی مورد استفاده قرار می‌گیرند تا بهترین تنظیمات الگوریتم‌ها و انتخاب بهینه ویژگی‌ها را براساس معیارهای سنجش عملکرد مشخص کنند.

[1] *Generalized Regression Neural Network*

[2] *Least-squares boosting*

در فرآیند تصمیم، انتخاب تابع برازندگی مناسب از اهمیت بالایی برخوردار است. با توجه به اینکه در این تحقیق ما دو فرایند دسته‌بندی و رگرسیون را داریم، بنابراین تابع برازندگی ترکیبی از این مرحله اتخاذ گردید تا بهترین عملکرد ممکن از هر دو فرایند حاصل شود. برای مرحله دسته‌بندی، ماکزیمم مقدار *F1-Score* به عنوان تابع برازندگی و برای مرحله رگرسیون نیز چون هدف تخمین قیمت است، لذا مقدار خطای تخمین به عنوان تابع برازندگی در نظر گرفته شده است. با توجه به آنچه که گفته شد، مسئله بهینه تحقیق حاضر بصورت زیر تعریف می‌شود:

$$max: \quad \sum_{i=1}^{4} W_i \, \overline{F1\text{-}Score}_i - \frac{\sum_{j=1}^{4} \mathrm{RMSE}_j}{\overline{\mathrm{price}}}$$

$$\text{s.t.} \quad \mathrm{LB} \le \mathrm{X}_{\mathrm{GA}} \le \mathrm{UB} \qquad (۱۴$$

$$\sum_{i=1}^{4} W_i = 1$$

$$\sum_{i=1}^{3} W_j = 1$$

در معادله (۱۴)، مقدار تابع $\sum_{i=1}^{4} W_i \, \overline{F1\text{-}Score}_i$ برابر با مقدار میانگین *F1-Score* هر سه کلاس و هر ۴ الگوریتم است. مقدار RMSE_j نیز برابر است با اختلاف بین قیمت واقعی و قیمت تخمینی هر یک از قیمت‌های چهارگانه و $\overline{\mathrm{price}}$ برابر با مقدار میانگین قیمت واقعی است. به عنوان مثال برای قیمت باز شدن داریم:

$$\mathrm{RMSE}_{Open} = \sqrt{\frac{\sum_{j=1}^{n} \mathrm{Open}_j - \sum_{i=1}^{3}\sum_{j=1}^{n} W_i \, \hat{\mathrm{Open}}_{ij}}{n}} \qquad (۱۵$$

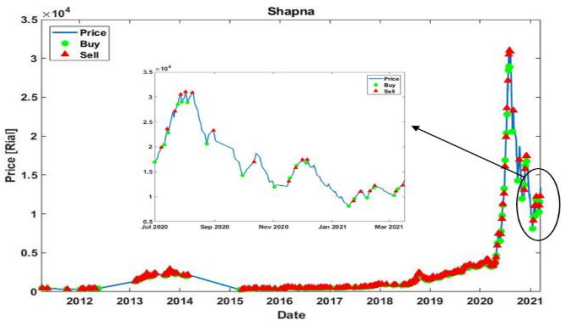
که در آن *Open* قیمت واقعی، $\hat{\mathrm{Open}}$ مقدار تخمینی و n تعداد نمونه‌ها (تعداد روزهای معاملاتی) است. علاوه بر این، X_{GA} در معادله (۱۴) برداری شامل تمامی ۵۵ متغیر تصمیم است. همچنین قیودی مبنی بر اینکه مجموع وزن‌های الگوریتم‌های دسته‌بندی و مجموع وزن‌های الگوریتم‌های رگرسیون برابر ۱ نیز برقرار است.

٤- یافته‌های تحقیق

۴-۱- برچسب‌گذاری

در فرآیند برچسب‌گذاری نقاط خرید و فروش روی نمودار قیمتی سهام، از روش پنجره لغزشی استفاده شده است که مقدار مناسب طول پنجره برای برچسب‌گذاری با استفاده از روش آنالیز حساسیت طول پنجره، بدست آمد. برای این منظور، طول پنجره را با کمترین مقدار برابر ۴ و بیشترین مقدار برابر ۳۰ در نظر گرفته شد. در انتخاب طول پنجره مناسب، دو پارامتر تعداد نقاط خرید و فروش و حداکثر بازدهی ممکن در نظر گرفته شد که بر اساس فرآیند برچسب‌گذاری مشخص شده در این بخش، نقاط خرید و فروش روی نمودار قیمتی سهام برچسب‌گذاری شد. در شکل ۳، نقاط خرید و فروش بر اساس فرایند برچسب‌گذاری با طول پنجره برابر ۵ نشان داده شده است. همانطور که از بخش زوم شده شکل مشخص است، الگوریتم برچسب‌گذاری به خوبی نقاط خرید و فروش را شناسایی کرده و برچسب مناسب به آن‌ها اختصاص داده است.

لازم به ذکر است که در الگوریتم کلی تحقیق، فرایند برچسب‌گذاری به صورت مداوم بر اساس قیمت‌های تخمینی از الگوریتم انجام و در نهایت، با بررسی بهترین نتیجه حاصل از الگوریتم، برچسب‌گذاری بهینه (برای قیمت بسته شدن تخمینی بهینه) نیز مشخص گردید.



شکل ۳- نقاط خرید و فروش کل دوره برای سهم شپنا.

جدول ۵- نتایج حاصل از برچسب‌گذاری

سهم	تعداد معاملات	تعداد روزهای نگهداشت	بازده معاملات الگوریتمی	بازده معاملات الگوریتمی (بدون کارمزد)	بازده خرید و نگهداشت
شپنا	۲۱۲	۷۳۴	۱۶۹۹۳۶۶۵	۲۰۶۶۳۰۴۰۳	۲۷۷۹
خودرو	۲۲۸	۹۱۰	۵۱۷۲۵۱۰۸۶	۱۷۲۴۶۸۱۹۶	۲۶۰۸
فجر	۱۳۰	۶۲۷	۹۶۶۳۶۱	۴۳۷۲۰۲۹	۱۲۷۲۵

بر اساس نتایج حاصل از برچسب‌گذاری، تعداد نقاط خرید و فروش (تعداد معاملات)، تعداد دوره‌های (روزهای) نگهداشت، بازده معاملات الگوریتمی و بازده خرید ابتدای دوره (ابتدای سال ۱۳۹۰) و فروش انتهای دوره (انتهای سال ۱۳۹۹) مطابق با جدول ۵ نشان داده شده است.

۴-۲- فرایند بهینه یکپارچه تخمین ـ دسته‌بندی

با انجام فرایند بهینه‌سازی یکپارچه تخمین ـ دسته‌بندی، نتایج بهینه حاصل گردید. پارامترهای بهینه الگوریتم‌ها به همراه وزن بهینه و ویژگی‌های بهینه انتخابی به ترتیب در جدول ۷ تا ۹ آورده شده است.

نمودار تخمین بهینه هر ۴ قیمت سهم شپنا به ازای ۱۵۰ رکورد آخر در شکل ۵ نشان داده شده است. همانطور که از این شکل مشخص است، قیمت‌های تخمینی حاصل از الگوریتم بهینه دارای دقت بالایی بخصوص برای قیمت بسته شدن است. برای سایر سهام نیز دقت بسیار مطلوبی حاصل گردید.

مقدار میانگین خطای درصدی تخمین برای هر سه سهم مورد مطالعه در جدول ۱۰ آورده شده است. نتایج دسته‌بندی کلاس‌های خرید، فروش و نگهداری (معاملات الگوریتمی) بر اساس معیارهای مورد مطالعه در جدول ۱۱ آورده شده است. بصورت میانگین، نتایج عملکردی بدست آمده از روش پیشنهادی حاکی از یک عملکرد دسته‌بندی بسیار مناسب است و نشان می‌دهد که روند بهینه‌سازی یکپارچه بخوبی قادر به بهینه‌سازی روند دسته‌بندی کلاس‌ها بوده است.

جدول ۷- مقادیر بهینه پارامتر الگوریتم‌ها

الگوریتم	پارامتر	کران پایین	کران بالا	مقدار بهینه		
				شپنا	خودرو	فجر
DTree	ماکزیمم تعداد شاخه‌ها	۱	۱۰۰	۱۴	۸	۱۱
	مینمم تعداد هر برگ در شاخه	۱	۲۰	۲	۲	۲
KNN	تعداد همسایگی	۱	۲۰	۳	۲	۲
RUSBoost	تعداد سیکل یادگیری	۱۰۰	۱۰۰۰	۲۳۷	۱۵۴	۲۹۶
	نرخ یادگیری	۰٫۰۱	۰٫۲	۰٫۰۵۶	۰٫۰۳۳	۰٫۰۶۹
RBN	مقدار گسترش[1] تابع شعاعی	۰٫۰۱	۱	۰٫۰۱۹	۰٫۰۱۳	۰٫۰۱۶
	ماکزیمم تعداد نرون‌ها	۱	۳۰	۱۸	۱۰	۱۶
LSTM	تعداد لایه‌های پنهان	۱۰	۱۰۰	۲۳	۲۹	۱۲
GRNN	مقدار گسترش تابع شعاعی	۰٫۰۱	۱	۰٫۰۳۹	۰٫۰۲۷	۰٫۰۴۰
LSBoost	تعداد سیکل یادگیری	۱۰۰	۱۰۰۰	۵۴۷	۳۶۲	۷۵۳
	نرخ یادگیری	۰٫۰۱	۰٫۲	۰٫۰۷۶	۰٫۰۳۲	۰٫۰۲۸

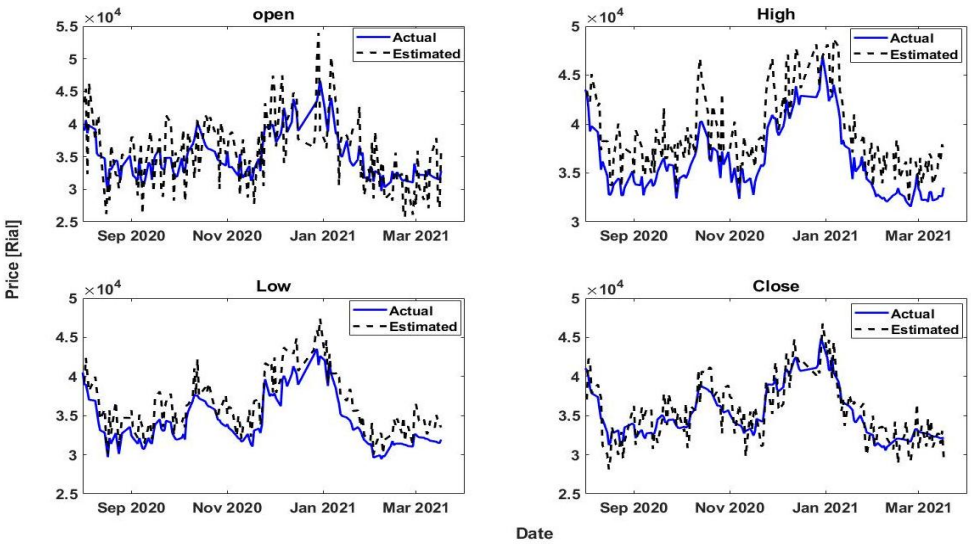
جدول ۸- وزن بهینه الگوریتم‌ها

مرحله	الگوریتم	کران پایین	کران بالا	مقدار بهینه		
				شپنا	خودرو	فجر
دسته‌بندی	*DTree*	۰	۱	۰٫۲۲۱	۰٫۲۴۵	۰٫۱۶۷
	KNN	۰	۱	۰٫۱۸۹	۰٫۲۱۲	۰٫۱۵۲
	RUSBoost	۰	۱	۰٫۲۷۴	۰٫۲۵۸	۰٫۲۹۹
تخمین	*ERBN*	۰	۱	۰٫۳۱۶	۰٫۲۸۵	۰٫۳۸۲
	LSTM	۰	۱	۰٫۵۲۷	۰٫۴۶۳	۰٫۵۷۴
	GRNN	۰	۱	۰٫۳۰۴	۰٫۳۲۷	۰٫۳۲۳
	LSBoost	۰	۱	۰٫۱۶۹	۰٫۲۱۰	۰/۱۰۳

[1] *Spread*

جدول ۹- ویژگی‌های بهینه منتخب

ویژگی	مقدار بهینه		
	شپنا	خودرو	فجر
قیمت	*Open* *High* *Low* *Close*	*Open* *High* *Low* *Close*	*Open* *High* *Low* *Close*
حجم	*Volume*	*Volume*	*Volume*
اندیکارتوهای غیرحجمی	*ADO* *SO12* *ACC9* *ACC12* *MOM12* *MMA12* *MACD* *RSI9* *RSI12* *Bollinger Upper12* *Bollinger Middle9* *ROC9* *WADLine*	*ADO* *SO9* *SO12* *ACC12* *MOM12* *MACD* *RSI9* *RSI12* *Bollinger Upper9* *ROC9* *ROC12* *PVI* *VROC12*	*SO9* *SO12* *ACC12* *MOM9* *MOM12* *MACD* *RSI12* *ROC9* *ROC12* *Bollinger Upper12* *PVT* *VROC9* *VROC12*
اندیکاتورهای حجمی	*NVI* *OBV* *VROC12*		
مجموع	۲۱	۱۸	۱۸



شکل ۵- قیمت‌های تخمینی بهینه برای سهم شپنا.

جدول ۱۰- درصد خطای میانگین تخمین بهینه قیمت‌ها

قیمت	میزان خطای میانگین تخمین (٪)		
	شپنا	خودرو	فجر
Open	۴٫۷۰۳	۴٫۰۲۸	۶٫۱۷۸
High	۸٫۱۵۹	۱۰٫۴۵۰	۶٫۲۴۸
Low	۶٫۱۸۱	۹٫۱۹۹	۵٫۵۳۵

Close	۳٫۱۰۵	۳٫۸۲۴	۵٫۴۳۱

جدول ۱۱- مقادیر بهینه معیارهای عملکردی دسته‌بندی کلاس‌ها

کلاس	معیار	مقدار		
		شپنا	خودرو	فجر
خرید	*ACC*	۰/۸۴۲	۰/۸۷۱	۰/۸۱۱
	Precision	۰/۹۰۱	۰/۹۴۱	۰/۸۳۶
	Recall	۰/۸۹۱	۰/۹۰۹	۰/۸۷۴
	F1-Score	۰/۸۹۶	۰/۹۲۵	۰/۸۵۵
فروش	*ACC*	۰/۸۶۳	۰/۸۶۳	۰/۸۳۰
	Precision	۰/۹۲۷	۰/۸۹۷	۰/۸۴۳
	Recall	۰/۹۱۱	۰/۸۹۹	۰/۸۱۳
	F1-Score	۰/۹۳۹	۰/۸۹۸	۰/۸۲۷
نگهداری	*ACC*	۰/۹۱۳	۰/۹۳۱	۰/۸۷۴
	Precision	۰/۹۳۹	۰/۹۵۶	۰/۹۱۱
	Recall	۰/۹۲۶	۰/۹۱۹	۰/۸۸۶
	F1-Score	۰/۹۴۳	۰/۹۳۷	۰/۸۹۸
میانگین	*ACC*	۰/۸۷۲	۰/۸۸۸	۰/۸۳۸
	Precision	۰/۹۲۲	۰/۹۲۰	۰/۸۶۳
	Recall	۰/۹۰۹	۰/۹۱۰	۰/۸۵۸
	F1-Score	۰/۹۲۶	۰/۹۲۶	۰/۸۶۰

۵- نتیجه‌گیری و پیشنهادات

با ظهور فرایند یادگیری عمیق (*DNN*)، عصر جدید از شبکه‌های عصبی مصنوعی ایجاد شد که توانایی‌های قابل توجهی در تشخیص الگو برای داده‌های زمانی طولانی دارند. در این پژوهش، یک سیستم یکپارچه معاملات الگوریتمی طراحی شد که علاوه بر تخمین چندقیمتی، بصورت همزمان قادر به دسته‌بندی سه کلاس خرید، فروش و نگهداری است. مدل‌های دسته‌بندی وابسته به تعداد و نوع ویژگی‌های مسئله هستند. با توجه به اینکه برای مسائل و داده‌های زمانی مختلف، ویژگی‌های مختلفی وجود دارد و نمی‌توان برای مسائل دسته‌بندی یک سری ویژگی‌های کلی برای تمامی مسائل در نظر گرفت، لذا ویژگی‌های انتخابی بسته به نتایج و تفسیر محقق دارد. در این پژوهش، تعداد ۴۰ ویژگی قیمتی، حجمی و اندیکاتورهای حجمی و غیرحجمی در نظر گرفته شده است. علاوه بر این، الگوریتم‌های دسته‌بندی و تخمین نیز دارای یک سری پارامتر هستند که تنظیم مقادیر آنها نیاز به تجربه و سعی و خطای مختلف دارد. بنابراین طراحی یک سیستمی که بصورت اتوماتیک این مقادیر را تنظیم کند، بسیار مناسب و جالب توجه است. یکی از بهترین رویکردها برای چنین مسائلی، استفاده از فرایند بهینه‌سازی با الگوریتم‌های فرا ابتکاری و در رأس آن الگوریتم ژنتیک است. این الگوریتم با داشتن فضای جستجو و تابع برازندگی، قادر است که مقادیر بهینه هایپرپارامترها را بدست آورد. بنابراین، بطور کلی می‌توان گفت که در صورت عدم استفاده از فرایند بهینه‌سازی (در اینجا با الگوریتم ژنتیک)، باید از همان تعداد ۴۰ ویژگی به اضافه پارامترهای از پیش تنظیم شدهٔ الگوریتم‌ها استفاده می‌شد که منجر به نتایج متفاوتی و گوناگونی (بسته به مجموعه مقادیر مختلف پارامترها) می‌شد. نتایج حاصل از این تحقیق، گویای آن است که مدل پیشنهادی داری عملکرد همزمان تخمین و دسته‌بندی بسیار مناسبی است و می‌تواند به عنوان یک ماژول معاملاتی خودکار در بازار ایران مورد استفاده قرار گیرد.

بر اساس نتایج حاصل از این پژوهش می‌توان پیشنهادات زیر را جهت تحقیقات آتی ارائه کرد:

- بررسی بازارهای جهانی از جمله کامودیتی‌ها، فارکس، ارزهای دیجیتال و بدست آوردن ویژگی‌های قمیتی، حجمی و اندیکاتورهای مشترک، جهت بنچمارک کردن ویژگی‌های مشترک انتخاب شده برای مسائل جهانی الگوریتم‌های معاملاتی.
- بازار سرمایه ایران یک بازار یک طرفه است ولی غالب بازارهای جهانی یک بازار دو طرفه هستند و در نتیجه روند برچسب‌گذاری آنها و انتخاب نقاط خرید، فروش و نگهداشت متفاوت از بازار سرمایه ایران است. بنابراین پیشنهاد می‌شود تا با استفاده از روش‌های مختلف برچسب گذاری در معاملات الگوریتمی، بهترین روش برچسب‌گذاری برای بازار سرمایه ایران و بازارهای جهانی ارائه شود.

منابع

اسماعیلی، وحید. ۱۴۰۰. ارائه سیستم معاملات الگوریتمی برای رمزارزها بر پایه یادگیری ماشین، پایان‌نامه کارشناسی ارشد. دانشکده مهندسی صنایع و سیستم‌ها. دانشگاه تربیت مدرس تهران.

چهرقانی، مجید. ۱۳۹۸. طراحی سیستم معاملاتی خودکار همراه با تعیین تعداد بهینه سهام جهت انجام معامله با استفاده از یادگیری Q عمیق. پایان‌نامه کارشناسی ارشد. دانشکده مهندسی صنایع و سیستم‌ها. دانشگاه تربیت مدرس.

ساعدی‌فر، خاطره. ۱۳۹۵. استراتژی انجام معاملات بهینه: رویکرد شبیه‌سازی عامل‌گرا. پایان‌نامه کارشناسی ارشد. دانشکده علوم مالی. دانشگاه خوارزمی.

موسوی‌انزهایی، سیدمجید. نیکومرام، هاشم. ۱۳۹۹. طراحی الگوی تعیین راهبردهای معاملاتی سهام با رویکرد مبتنی بر آینده‌پژوهی. تحلیل بنیادی، مهندسی ویژگی‌ها و الگوریتم‌های یادگیری ماشین. مهندسی مالی و مدیریت اوراق بهادار. شماره ۴۵. ۵۱۷-۴۹۹.

هروانی، محمد. و خلیلی‌عراقی، مریم. ۱۳۹۹. طراحی استراتژی معاملات الگوریتمی با معرفی اندیکاتور میانگین متحرک تعدیل‌پذیر (AMA) جهت پیش‌بینی حرکت قیمت سهام در آینده در بازار سرمایه ایران. اولین کنفرانس بین‌المللی چالش‌ها و راهکارهای نوین در مهندسی صنایع و مدیریت و حسابداری.

یافتیان، امیرحسین. ۱۳۹۷. طراحی یک سیستم معاملات الگوریتمی با استفاده از پردازش تصویر به وسیله شبکه عصبی کانولوشن، پایان‌نامه کارشناسی ارشد. دانشکده مهندسی صنایع و سیستم‌ها. دانشگاه تربیت مدرس.

AYCEL, Ü., & SANTUR, Y. (2022). A new algorithmic trading approach based on ensemble learning and candlestick pattern recognition in financial assets. Turkish Journal of Science and Technology, 17(2), 167-184.

Cao, X., Francis, A., Pu, X., Zhang, Z., Katsikis, V., Stanimirovic, P., ... & Li, S. (2023). A novel recurrent neural network based online portfolio analysis for high frequency trading. Expert Systems with Applications, 120934.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Christodoulaki, E., Kampouridis, M., & Kyropoulou, M. (2023). Enhanced Strongly typed Genetic Programming for Algorithmic Trading.

Cohen, G. (2023). Technical Analysis in Investing. Review of Pacific Basin Financial Markets and Policies (RPBFMP), 26(02), 1-23.

Conti, J. P. J., & Lopes, H. S. (2019). Algorithmic Trading Using Genetic Algorithms in the Brazilian Stock Exchange.

Ding, G., & Qin, L. (2020). Study on the prediction of stock price based on the associated network model of LSTM. International Journal of Machine Learning and Cybernetics, 11(6), 1307-1317.

Eiamkanitchat, N., Moontuy, T., & Ramingwong, S. (2017). Fundamental analysis and technical analysis integrated

Conference on Artificial Intelligence, Cuvu, Yanuca Island, Fiji, August 26-30, 2019, Proceedings, Part III 16 (pp. 748-754). Springer International Publishing.

Moghaddam, A. H., Moghaddam, M. H., & Esfandyari, M. (2016). Stock market index prediction using artificial neural network. Journal of Economics, Finance and Administrative Science, 21(41), 89-93.

Nagaraj, P., Muneeswaran, V., Reddy, T. D., Reddy, B. V. G., Reddy, T. G., & Suresh, P. (2023, January). Automated Stock Price Prediction Using LSTM-ANN. In 2023 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-9). IEEE.

Ponomarev, E. S., Oseledets, I. V., & Cichocki, A. S. (2019). Using reinforcement learning in the algorithmic trading problem. Journal of Communications Technology and Electronics, 64, 1450-1457.

Rundo, F. (2019). Deep LSTM with reinforcement learning layer for financial trend prediction in FX high frequency trading systems. Applied Sciences, 9(20), 4460.

Seighaly, M., Koosha, E., & Abbasi, E. (2023). Measuring the Accuracy and Precision of Random Forest, Long Short-Term Memory, and Recurrent Neural Network Models in Predicting the Top and Bottom of Bitcoin price. Journal of Mathematics and Modeling in Finance, 2(2), 107-128.

Seyfert, R. (2018). Automation and affect: A study of algorithmic trading. In Affect in Relation (pp. 197-217). Routledge.

Sezer, O. B., Ozbayoglu, A. M., & Dogdu, E. (2017, April). An artificial neural network-based stock trading system using technical analysis and big data framework. In proceedings of the southeast conference (pp. 223-226).

Shavandi, A., & Khedmati, M. (2022). A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets. Expert Systems with Applications, 208, 118124.

system for stock filtration. Cluster Computing, 20, 883-894.

Felizardo, L. K., Paiva, F. C. L., de Vita Graves, C., Matsumoto, E. Y., Costa, A. H. R., Del-Moral-Hernandez, E., & Brandimarte, P. (2022). Outperforming algorithmic trading reinforcement learning systems: A supervised approach to the cryptocurrency market. Expert Systems with Applications, 202, 117259.

Germanov, S. S. (2023). Theoretical Methodological Justification in Technical Analysis of Financial Markets. Knowledge: International Journal, 58(1).

Huang, Y., Capretz, L. F., & Ho, D. (2019, May). Neural network models for stock selection based on fundamental analysis. In 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE) (pp. 1-4). IEEE.

Huang, Z., Gong, W., & Duan, J. (2023). TBDQN: A novel two-branch deep Q-network for crude oil and natural gas futures trading. Applied Energy, 347, 121321.

Khandelwal, V., & Tak, S. (2022). Algorithmic Trading on Trading Systems. NOLEGEIN-Journal of Global Marketing, 5(1).

Kissell, R. (2021). Algorithmic trading methods: Applications using advanced statistics, optimization, and machine learning techniques. Academic Press.

Lei, K., Zhang, B., Li, Y., Yang, M., & Shen, Y. (2020). Time-driven feature-aware jointly deep reinforcement learning for financial signal representation and algorithmic trading. Expert Systems with Applications, 140, 112872.

Liu, P., Zhang, Y., Bao, F., Yao, X., & Zhang, C. (2023). Multi-type data fusion framework based on deep reinforcement learning for algorithmic trading. Applied Intelligence, 53(2), 1683-1706.

Mazumdar, K., Zhang, D., & Guo, Y. (2019). Multi-peak Algorithmic Trading Strategies Using Grey Wolf Optimizer. In PRICAI 2019: Trends in Artificial Intelligence: 16th Pacific Rim International

Cryptocurrency. Journal of Computer Information Systems, 60(6), 555-568.
Wen, Y., & Yuan, B. (2018, March). Use CNN-LSTM network to analyze secondary market data. In Proceedings of the 2nd International Conference on Innovation in Artificial Intelligence (pp. 54-58).
Tuininga, F. (2023). Uncovering the Potential of Deep Learning in Algorithmic Trading: Deep reinforcement learning stock market trading, utilizing a CNN with candlestick images (Master's thesis, University of Twente).
Vo, A., & Yost-Bremm, C. (2020). A high-frequency algorithmic trading strategy for

Design an integrated optimal hybrid algorithmic trading system with simultaneous multi-price estimation

Shiva Ghasempour[۱]
Shadi Shahverdiani[*۲]
Amirreza Keyghobadi[۳]
Mahdi Madanchi Zaj[۴]

Abstract

Designing automated trading systems has gained significant attention in various countries, including Iran. The advantage of such systems lies in the increased speed and accuracy of decision-making without emotional interference. Having a model that can simultaneously estimate the opening, highest, lowest, and closing prices is of great benefit to investors. With the emergence of deep learning processes and artificial neural networks, the capabilities of pattern recognition for long-term time series data have significantly improved. In this research, a unified algorithmic trading system was designed using a deep neural network called LSTM (Long Short-Term Memory) to simultaneously estimate the four prices and classify the points into three classes: buy, sell, and hold. Various classification algorithms were used, and an optimized integrated algorithmic trading system was designed using a genetic algorithm. In the

[۱] *PhD. Student in Financial Engineering, Department of Financial Management, Faculty of Management and Economy, Science and Research Branch, Islamic Azad University, Tehran, Iran. (shivaghasempour@gmail.com)*
[۲] *Assistant Professor, Department of Financial Management, Faculty of Humanities, Shahr-e-Qods Branch, Islamic Azad University, Tehran, Iran. and Visiting professor of Science and Research Branch, Islamic Azad University, Tehran, Iran. (Corresponding author) (shshahverdiani@gmail.com)*
[۳] *Department of Accounting, Faculty of Economics and Accounting, Central Tehran Branch, Islamic Azad University, Tehran, Iran. (a.keyghobadi@iauctb.ac.ir)*
[۴] *Department of Financial Management, Faculty of Management, Electronic Campus, Islamic Azad University, Tehran, Iran. (madanchi@iauec.ac.ir)*

classification process, ensemble learning methods including decision tree algorithms, K-Nearest Neighbors (KNN), RUSBoost, and Radial Basis Function Neural Network (RBFNN) were used. Since different problems and time-series data require different features, it is essential to select the appropriate features based on the researcher's interpretation and results. In this study, 40 price-related features, including volume-based indicators, were considered, and the optimal features were selected simultaneously through the optimization process. To evaluate the proposed model, data from Shapna, Khodro, and Fajr stock market in Iran from 2012 to 2020 were used, and the MATLAB software was employed. The results of this research showed that the multi-price prediction algorithm achieved an accuracy of over 90% for all four prices. Furthermore, the optimal values of performance metrics for classification of the three classes (buy, sell, and hold) exhibited high accuracy, indicating the credibility and effectiveness of the proposed method in simultaneous estimation and classification. Therefore, the proposed approach can be considered highly suitable for automated trading and can be implemented in designing and executing algorithmic trading robots in the Iranian stock market.

Keywords: *Algorithmic trading, Deep neural networks, LSTM, Optimization, Multi-price estimation.*