**پیش‌بینی قیمت سهام با استفاده از مدل‌های LSTM، GRU و ARIMA**

**مقایسه و بررسی عملکرد**

**عارفه عمیدیان، سیده بیتا امیری و زهره فصیح فر**

**دانشگاه حکیم سبزواری**

**چکیده**

این مطالعه به بررسی کارایی مدل‌های مختلف یادگیری ماشین و مدل‌های آماری می‌پردازد که توجه تحلیلگران مالی را جلب کرده‌اند. دو مورد از این مدل‌ها، حافظه بلندمدت-کوتاه‌مدت (LSTM) و واحد بازگشتی گیت‌دار (GRU) هستند که از انواع شبکه‌های عصبی بازگشتی محسوب می‌شوند، در حالی که مدل میانگین متحرک خودرگرسیو یکپارچه (ARIMA) یک مدل آماری است. به دلیل اینکه تأثیر متغییرهای دیگری را نیز در پیش بینی میخواهیم لحاظ کنیم، ARIMAX گزینه بهتری است. این مدل‌ها برای پیش‌بینی داده‌های بازار سهام در بخش‌های مختلف اقتصادی مورد استفاده قرار خواهند گرفت. در چشم‌انداز پویا و متغیر بازارهای مالی، پیش‌بینی دقیق از اهمیت بالایی برخوردار است. این مقاله به حوزه مالی کمی کمک می‌کند و یک تحلیل مقایسه‌ای جامع از این مدل‌ها بر روی داده‌های تاریخی بازار سهام در سه بخش: استخراج، تولید و خدمات (که به ترتیب به عنوان بخش‌های اولیه، ثانویه و ثالث اقتصاد در نظر گرفته می‌شوند) انجام می‌دهد. عملکرد مدل‌ها با استفاده از میانگین مربعات خطا (MSE) بر روی سهام بانک صادرات ایران که نمایانگر این بخش‌ها هستند، ارزیابی می‌شود. نتایج نشان‌دهنده قدرت شبکه‌های عصبی بازگشتی در شناسایی الگوهای پیچیده است. علاوه بر این، نتایج بررسی خواهند کرد که آیا کارایی هر مدل تحت تأثیر بخش اقتصادی که داده‌های آن پیش‌بینی می‌شود، قرار می‌گیرد یا خیر.

**مقدمه**

در سال‌های اخیر، توجه محققان و فعالان حوزه مالی به سمت پویایی بازارهای مالی و پیچیدگی نوسانات قیمت سهام افزایش یافته است. با ورود سرمایه به بازار بورس، سرمایه‌گذاران به دنبال سهامی با بازدهی و سودآوری بالا هستند تا سرمایه‌گذاری موفقی داشته باشند. نوسانات قیمت سهام ناشی از عرضه و تقاضای دائمی آن در بازار، پیچیدگی‌های جدیدی برای پیش‌بینی قیمت‌ها ایجاد کرده است. پیش‌بینی صحیح قیمت سهام به سرمایه‌گذاران کمک می‌کند که دید کلی و مناسبی نسبت به آینده سهام داشته باشند و تصمیمات بهتری بگیرند.

از این رو، مدل‌های پیش‌بینی به ابزاری کلیدی برای تحلیل‌گران مالی تبدیل شده‌اند که تلاش می‌کنند با استفاده از داده‌های گذشته، رفتار آینده بازار را درک و پیش‌بینی کنند. در بازار بورس برای تصمیم‌گیری‌های خرید و فروش، تجزیه و تحلیل قیمت سهام با استفاده از روش‌های مختلف ضروری است. این روش‌ها شامل تحلیل تکنیکال، که بر داده‌ها و نمودارهای قیمتی تمرکز دارد، و تحلیل فاندامنتال، که ارزش ذاتی سهام را از طریق بررسی عوامل خارجی تخمین می‌زند، هستند.

تحلیل تکنیکال بر شناسایی روندهای گذشته برای پیش‌بینی رفتار آینده تمرکز دارد و از این رو، ابزارهای یادگیری ماشین و مدل‌های سری زمانی نیز در این زمینه نقش مهمی ایفا می‌کنند. در میان این مدل‌ها، شبکه‌های عصبی بازگشتی (RNN) و مدل‌های پیشرفته‌تری مانند GRU (Gated Recurrent Unit) و LSTM (Long Short-Term Memory) به دلیل توانایی در شناسایی الگوهای بلندمدت و غیرخطی در داده‌ها، جایگاه ویژه‌ای در پیش‌بینی قیمت سهام پیدا کرده‌اند. GRU به دلیل ساختار ساده‌تر و زمان آموزش کوتاه‌تر خود مورد توجه قرار گرفته است، در حالی که LSTM به دلیل حافظه پیچیده‌تر، در شناسایی الگوهای طولانی‌مدت عملکرد بهتری دارد.

علاوه بر مدل‌های مبتنی بر یادگیری عمیق، یکی از روش‌های کلاسیک و محبوب در تحلیل سری‌های زمانی، مدل خودرگرسیو میانگین متحرک یکپارچه (ARIMA) است. ARIMA به دلیل ساختار خطی خود، بیشتر برای داده‌هایی با الگوهای ساده‌تر و بدون نوسانات شدید مورد استفاده قرار می‌گیرد. این مدل بر مبنای روابط گذشته بین مقادیر یک سری زمانی کار می‌کند و پیش‌بینی‌های آن مبتنی بر روندهای مشاهده‌شده و خودرگرسیو است. با این حال، در مقایسه با مدل‌های پیچیده‌تر مانند GRU و LSTM، ARIMA در مواجهه با داده‌های پیچیده‌تر ممکن است محدودیت‌هایی داشته باشد، اما همچنان در پیش‌بینی سری‌های زمانی مالی مفید است.

در این مقاله، اثربخشی سه مدل پیش‌بینی برجسته، یعنی شبکه حافظه کوتاه‌مدت بلندمدت (LSTM)، واحد بازگشتی دروازه‌ای (GRU) و مدل خودرگرسیو میانگین متحرک یکپارچه (ARIMA)، برای پیش‌بینی داده‌های بازار سهام بررسی شده است. LSTM و GRU به دلیل توانایی خود در تحلیل داده‌های سری زمانی و شناسایی الگوهای غیرخطی، در پیش‌بینی‌های مالی نوید زیادی دارند، در حالی که ARIMA همچنان به عنوان یک روش قابل‌اعتماد و کارآمد در پیش‌بینی سری‌های زمانی خطی شناخته می‌شود. بررسی دقیق این مدل‌ها می‌تواند به بهبود استراتژی‌های سرمایه‌گذاری و توسعه مالی کمک کند.

در این مطالعه، عملکرد دو مدل شبکه عصبی بازگشتی، GRU (Gated Recurrent Unit) و LSTM (Long Short-Term Memory)، برای پیش‌بینی داده‌های سری زمانی مورد بررسی قرار گرفته است. هر دو مدل برای حل مشکل یادگیری روابط بلندمدت در داده‌های سری زمانی طراحی شده‌اند، اما GRU ساختار ساده‌تری نسبت به LSTM دارد که منجر به کاهش زمان آموزش و نیاز به حافظه می‌شود. در حالی که LSTM از حافظه داخلی پیچیده‌تری برای نگهداری اطلاعات استفاده می‌کند، GRU در بسیاری از موارد دقت مشابهی با کارایی بالاتر نشان می‌دهد.هدف اصلی مقاله ما تعیین بهترین الگوریتم آموزش‌دیده برای پیش‌بینی موثر ارزش‌های آتی پرتفوی ما با استفاده از الگوریتم‌های یادگیری ماشین مبتنی بر LSTM و GRU برای پیش‌بینی قیمت‌های بسته شدن تعدیل‌شده مجموعه دارایی‌ها است.

**روش ها**

دیتاست 14 ساله (2023-2009)سهام بانک صادرات ایران(وبصادر) با فرمت csv جمع آوری شده که قیمت نهایی و بسته شده سهام را در قاب 14 سال زمانی نمایش می دهد.این داده ها را از سایت [مدیریت فناوری بورس تهران](https://www.tsetmc.com/) دانلود کردیم.

این مدل ها را در Google Colab اجرا کرده ایم.از کتابخانه های مختلف پایتون مثل ، statsmodels ، sklearn ، numpy ، matplotlib ، pandas و keras استفاده کردیم.ما پیش پردازش های مختلفی روی داده ها انجام دادیم.ابتدا یک ستون جدید به داده ها اضافه کردیم به نام درصد تغییرات و در آن تغییرات درصدی ستون قیمت بسته شدن را قرار داده ایم. اختلاف نسبی هر مقدار ستون قیمت بسته شدن را نسبت به مقدار قبلی در همان ستون حساب می‌کند و ضرب در صد میشود تا به درصد تبدیل شود.

ابتدا یک ستون جدید به داده‌ها اضافه کردیم به نام "درصد تغییرات" و در آن تغییرات درصدی ستون قیمت بسته شدن را قرار دادیم. اختلاف نسبی هر مقدار ستون قیمت بسته شدن نسبت به مقدار قبلی در همان ستون حساب شد و ضرب در صد شد تا به درصد تبدیل شود.

با این حال، پس از بررسی ماتریس همبستگی، مشاهده کردیم که این ویژگی هیچ ارتباط معناداری با سایر ستون‌های موجود ندارد. به همین دلیل، تصمیم گرفتیم این ستون را از فرآیند مدل‌سازی حذف کنیم. این تحلیل نشان می‌دهد که حتی اضافه کردن ویژگی‌های ظاهراً منطقی نیز همیشه منجر به بهبود مدل یا نتایج نمی‌شود.

A screenshot of a computer screen

Description automatically generated

در مرحله تمیزسازی داده‌ها (Data Cleaning)، بررسی کردیم که هیچ داده تکراری و هیچ داده از دست‌رفته‌ای در مجموعه داده‌ها وجود ندارد. همچنین داده‌های پرت (Outliers) را شناسایی کرده و آن‌ها را با مقدار میانگین جایگزین کردیم. پس از 5 مرحله محاسبه و جایگزینی میانگین، تعداد داده‌های پرت از 86 عدد به صفر کاهش یافت.

سپس برای تحلیل سری زمانی قیمت بسته شدن، از تابع seasonal\_decompose موجود در کتابخانه **statsmodels** استفاده کردیم. این تابع سری زمانی را به سه جزء اصلی تجزیه می‌کند:

1. **روند(Trend):**تغییرات کلی و بلندمدت در داده‌ها که نشان‌دهنده مسیر کلی سری زمانی است.
2. **فصلی بودن(Seasonality):**الگوهای تکرارشونده با یک بازه زمانی مشخص. در اینجا با فرض ماهانه بودن الگوها، بازه زمانی (period) برابر با 30 روز در نظر گرفته شد.
3. **باقیمانده(Residual):** بخش غیرقابل توضیح داده‌ها که به‌عنوان نویز یا تغییرات تصادفی در نظر گرفته می‌شود.

A diagram of a graph

Description automatically generated with medium confidence

ما از مدل **جمعی (Additive)** استفاده کردیم که بر اساس آن، سری زمانی به صورت زیر تجزیه می‌شود:

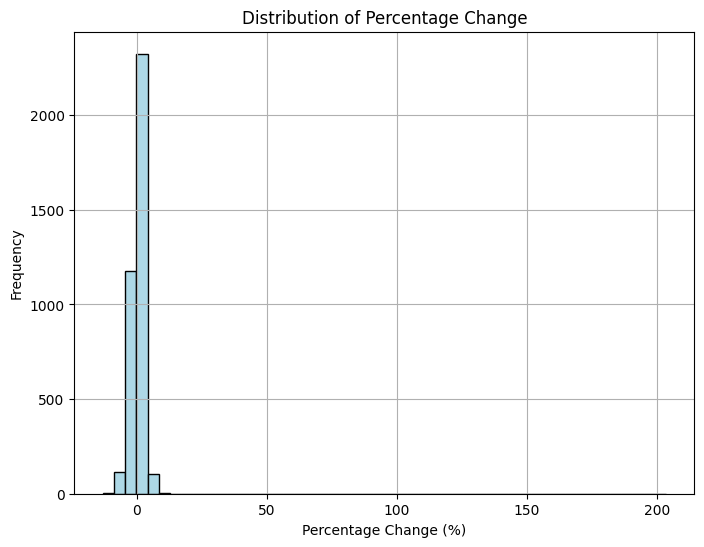
**Y(t)=Trend(t)+Seasonality(t)+Residual(t)**

در این معادله، **Y(t)** نشان‌دهنده مقدار سری زمانی در زمان **t**است که از مجموع سه جزء روند، فصلی بودن و باقیمانده تشکیل شده است. این تحلیل به ما کمک کرد تا الگوهای پنهان در داده‌ها را شناسایی کرده و درک بهتری از ساختار سری زمانی به‌دست آوریم.

برای بهبود عملکرد مدل‌های یادگیری ماشین و کاهش تأثیر مقیاس‌های مختلف ویژگی‌ها، عملیات **مقیاس‌بندی (Scaling)** روی ویژگی‌های عددی داده‌ها انجام شد. این عملیات تضمین می‌کند که تمامی ویژگی‌ها در یک بازه یکسان قرار گرفته و مدل‌ها با دقت بیشتری آموزش داده شوند. برای این منظور از ابزار MinMaxScaler موجود در کتابخانه scikit-learn استفاده شد. این ابزار داده‌ها را با استفاده از فرمول زیر به بازه [0, 1] نرمال‌سازی می‌کند:

ویژگی‌های عددی شامل close، open، high، low و vol برای مقیاس‌بندی انتخاب شدند. سایر ستون‌ها مانند تاریخ (date) و درصد تغییرات (percentage\_change) به دلیل ماهیت غیر عددی یا قرار گرفتن در مقیاس مناسب، از این فرآیند حذف شدند.

این پیش‌پردازش به مدل‌ها کمک می‌کند تا تأثیر مقیاس‌های متفاوت کاهش یافته و نتایج نهایی بهبود یابد.



**مدل LSTM**

**مدل‌سازی پیشرفته سری‌های زمانی با GRU و LSTM**

در این مقاله، به بررسی کاربرد مدل‌های یادگیری عمیق GRU و LSTM در پیش‌بینی سری‌های زمانی، به ویژه قیمت سهام، می‌پردازیم. این مدل‌ها به دلیل توانایی در یادگیری وابستگی‌های طولانی‌مدت، ابزاری قدرتمند برای تحلیل داده‌های مالی محسوب می‌شوند.

**مقایسه عملکرد GRU و LSTM در پیش‌بینی قیمت سهام**

در این پژوهش، به مقایسه عملکرد دو مدل محبوب یادگیری عمیق، یعنی GRU و LSTM، در پیش‌بینی قیمت سهام می‌پردازیم. هر دو مدل به دلیل توانایی در حفظ اطلاعات گذشته، ابزار مناسبی برای تحلیل سری‌های زمانی هستند. با این حال، تفاوت‌های معماری آن‌ها منجر به تفاوت در عملکردشان می‌شود.

**پیش‌پردازش داده‌ها با پانداس**

قبل از آموزش مدل، داده‌های تاریخی قیمت سهام با استفاده از کتابخانه پانداس پاکسازی، نرمال‌سازی و به ویژگی‌های مناسب تبدیل می‌شوند. سپس، داده‌ها به دو مجموعه آموزشی و آزمایشی تقسیم می‌شوند تا عملکرد مدل به صورت دقیق ارزیابی شود.

**تفاوت‌های GRU و LSTM**

هر دو مدل GRU و LSTM برای حل مشکل محوشدگی گرادیان در شبکه‌های عصبی بازگشتی (RNN) طراحی شده‌اند. با این حال، مکانیسم‌های دروازه‌ای آن‌ها تفاوت‌هایی دارد. LSTM از سه دروازه (فراموشی، ورودی و خروجی) برای کنترل جریان اطلاعات استفاده می‌کند، در حالی که GRU از دو دروازه (به‌روزرسانی و بازنشانی) بهره می‌برد. این تفاوت‌ها بر پیچیدگی مدل و توانایی آن در یادگیری وابستگی‌های طولانی‌مدت تأثیر می‌گذارند.

LSTM و GRU دو نوع از شبکه‌های عصبی بازگشتی هستند که برای مدل‌سازی داده‌های سری زمانی بسیار مناسب هستند. هر دو مدل از مکانیسم‌های دروازه‌ای برای کنترل جریان اطلاعات استفاده می‌کنند. با این حال، ساختار داخلی و نحوه عملکرد دروازه‌ها در این دو مدل متفاوت است. این تفاوت‌ها باعث می‌شود که LSTM و GRU در برخی از مسائل عملکرد متفاوتی داشته باشند.

**استراتژی آموزش و ارزیابی**

برای آموزش مدل، از روش‌های بهینه‌سازی مانند Adam یا RMSprop استفاده می‌شود. پس از آموزش، عملکرد مدل با استفاده از معیارهایی مانند RMSE (میانگین مربعات خطا) و MAE (میانگین قدر مطلق خطا) بر روی مجموعه آزمایشی ارزیابی می‌شود.

در شبکه‌های عصبی، هر نورون به عنوان یک واحد محاسباتی عمل می‌کند. این نورون‌ها به صورت لایه به لایه سازماندهی شده‌اند. اطلاعات ورودی (مثلاً قیمت‌های تاریخی سهام) به لایه ورودی وارد شده و سپس به لایه‌های بعدی منتقل می‌شود. در هر لایه، نورون‌ها با انجام محاسبات ساده، اطلاعات را پردازش می‌کنند. در نهایت، لایه خروجی یک عدد به عنوان پیش‌بینی قیمت سهام ارائه می‌دهد.

برای آموزش شبکه عصبی، از داده‌های تاریخی استفاده می‌شود. هدف آموزش این است که شبکه عصبی بتواند الگوهای موجود در داده‌ها را یاد گرفته و از آن‌ها برای پیش‌بینی آینده استفاده کند. در این فرایند، پارامترهای شبکه(وزن ها و بایاس) به گونه‌ای تنظیم می‌شوند که خطای پیش‌بینی به حداقل برسد.

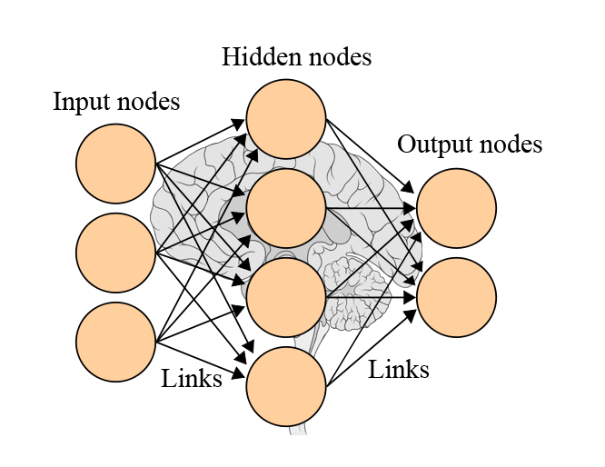
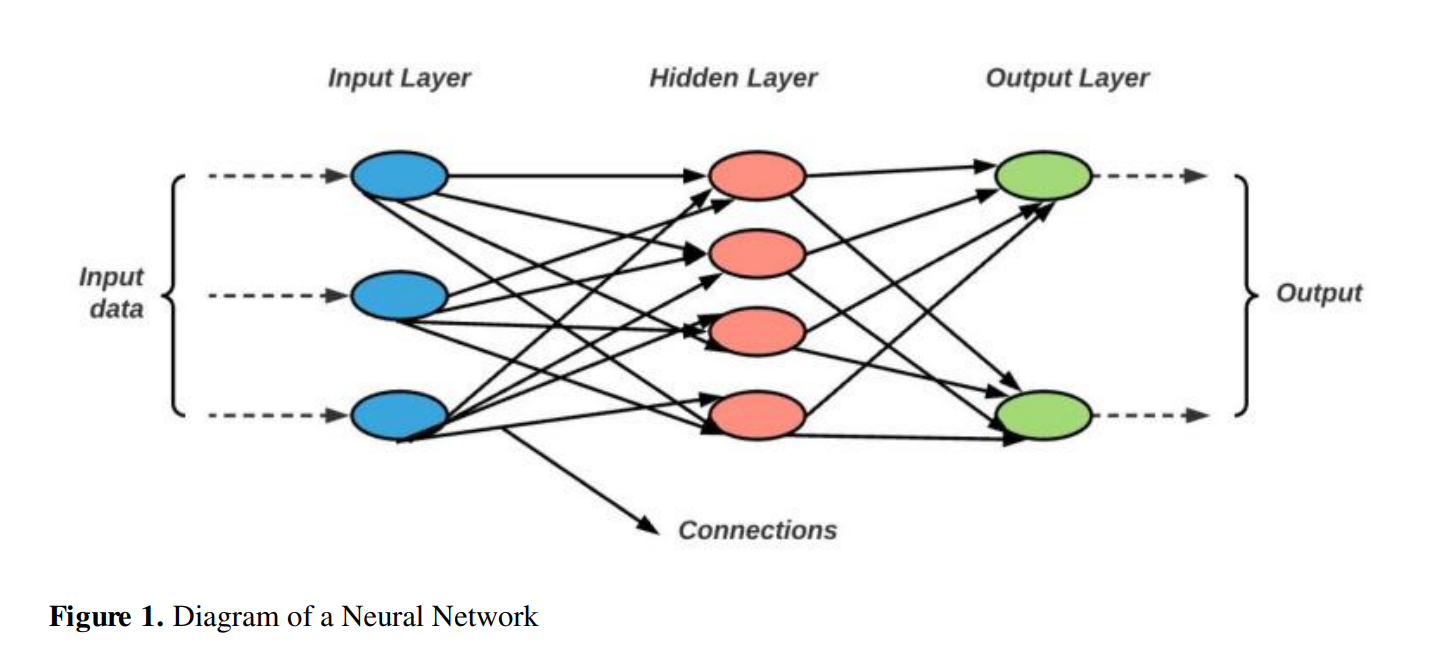


Figure 1. Diagram of a Neural Network



آموزش شبکه عصبی، فرآیندی است که طی آن پارامترهای شبکه (وزن‌ها و بایاس‌ها) به گونه‌ای تنظیم می‌شوند که شبکه بتواند الگوهای موجود در داده‌ها را یاد گرفته و پیش‌بینی‌های دقیقی انجام دهد. در واقع، آموزش به شبکه می‌آموزد که چگونه عملیات ریاضی پیچیده‌ای را بر روی داده‌های ورودی انجام دهد تا به خروجی مطلوب برسد.

قبل از شروع آموزش، داده‌ها باید پیش‌پردازش شوند. پیش‌پردازش شامل مراحل مختلفی مانند نرمال‌سازی، استانداردسازی و ایجاد ویژگی‌های جدید است. هدف از پیش‌پردازش، آماده‌سازی داده‌ها و تبدیل داده‌ها به فرمتی برای آموزش شبکه عصبی است.

در طول آموزش، شبکه عصبی مکررا پیش‌بینی‌هایی انجام می‌دهد و سپس خطای بین این پیش‌بینی‌ها و مقادیر واقعی را محاسبه می‌کند. با استفاده از الگوریتم‌های بهینه‌سازی، وزن‌ها و بایاس‌ها به گونه‌ای تنظیم می‌شوند که این خطا به حداقل برسد.

در طی فرایند آموزش، مدل شبکه عصبی با تنظیم وزن‌ها و بایاس‌های خود، الگوهای پیچیده‌ای را در داده‌های آموزشی شناسایی کرده است. این الگوها به مدل اجازه می‌دهند تا روابط بین داده‌های مختلف را درک کرده و از آن‌ها برای پیش‌بینی آینده استفاده کند. اکنون، با استفاده از داده‌های آزمایشی می‌توانیم عملکرد مدل را در پیش‌بینی سری‌های زمانی جدید ارزیابی کنیم.

به دلیل تفاوت در معماری سلول‌های LSTM و GRU، هر یک از این شبکه‌های عصبی بازگشتی روشی منحصر به فرد برای پردازش اطلاعات و یادگیری الگوها دارند.

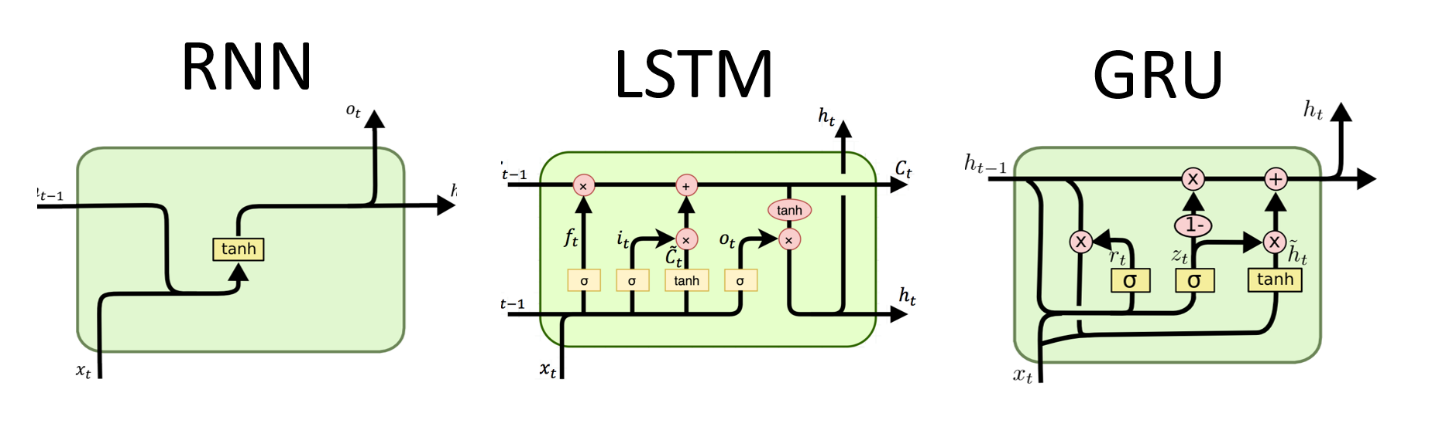


Figure 2. Diagram of architecture of LSTM, GRU, and standard RNN

یکی از چالش‌های اصلی در شبکه‌های عصبی بازگشتی ساده، مشکل ناپدید شدن گرادیان است. LSTM و GRU برای مقابله با این مشکل، مکانیسم‌های دروازه‌ای را معرفی کردند. این دروازه‌ها به شبکه اجازه می‌دهند تا اطلاعات مهم را برای مدت طولانی‌تری حفظ کرده و الگوهای پیچیده‌تر در داده‌ها را شناسایی کند. دروازه فراموشی و دروازه خروجی در LSTM و دروازه به روز رسانی و تنظیم مجدد در GRU به این شبکه‌ها امکان می‌دهند تا اطلاعات مرتبط را انتخاب کرده و اطلاعات نامرتبط را حذف کنند. در مقابل، شبکه‌های عصبی بازگشتی ساده به دلیل عدم وجود چنین مکانیسم‌هایی، تمایل به فراموش کردن اطلاعات گذشته دارند.

کتابخانه Keras ابزاری قدرتمند برای ساخت مدل‌های یادگیری عمیق از جمله مدل‌های LSTM و GRU است. با استفاده از Keras، می‌توانیم به سادگی یک مدل شبکه عصبی بازگشتی طراحی کنیم و آن را بر روی داده‌های سری زمانی آموزش دهیم. برای این کار، کافی است لایه‌های LSTM یا GRU را به مدل اضافه کرده و سپس از توابع model.fit برای آموزش و model.predict برای پیش‌بینی استفاده کنیم. در نهایت، می‌توانیم با استفاده از کتابخانه Matplotlib نموداری رسم کنیم تا تفاوت بین مقادیر پیش‌بینی شده و مقادیر واقعی را مقایسه کنیم.

انتخاب بهترین مدل برای پیش‌بینی قیمت سهام بسیار مهم است. به همین دلیل، از معیار میانگین مربعات خطا (MSE) برای مقایسه عملکرد مدل‌های مختلف استفاده شده است. همچنین، برای بررسی اینکه آیا مدلی خاص برای بخش‌های مختلف اقتصادی مناسب است یا خیر، مدل‌ها را روی داده‌های مربوط به سه بخش تولید، خدمات و کشاورزی آزمایش شده اند. نتایج این آزمایش‌ها به ما کمک می‌کند تا بهترین مدل را برای هر بخش اقتصادی انتخاب کنیم.

**مدل ARIMA**

مدل ARIMA یک مدل آماری بدون استفاده از یادگیری ماشین سنتی است، به این معنی که شخص باید سه پارامتر را وارد کند و سپس می‌تواند مدل را اجرا کند. با این حال، ابتدا داده ها باید ثابت در نظر گرفته شوند تا مدل ARIMA به طور موثر عمل کند. داده های ثابت به این معنی است که واریانس داده ها ثابت می ماند و داده ها حول یک مقدار میانگین نوسان می کنند. فرآیند تفاوت این مشکل را حل می کند. اگر 𝑡𝑡 نشان دهنده زمان، متغیر مستقل، و 𝑦𝑦(𝑡𝑡) نشان دهنده متغیر وابسته باشد، فرآیند تفاضل به صورت زیر انجام می شود. برای اینکه مدل به درستی مقادیر سری زمانی را برای سهام هدف پیش بینی کند، از داده های آموزشی برای برازش مدل و تنظیم دقیق پارامترهای آن استفاده می شود. از مدل خواسته می شود تا پس از پایان آموزش، قیمت سهام را برای روزهای بعد از مجموعه داده آموزشی پیش بینی کند. داده های تست، که قیمت واقعی سهام است، سپس با این پیش بینی ها مقایسه می شود. عملکرد این مدل ها در پاراگراف های بعدی توضیح داده خواهد شد.

مدل ARIMA (AutoRegressive Integrated Moving Average) یک ابزار آماری پرکاربرد برای تحلیل و پیش‌بینی سری‌های زمانی است. این مدل به دلیل سادگی و کارایی، به طور گسترده‌ای در حوزه‌های مختلفی مانند اقتصاد، مالی، هواشناسی و مهندسی مورد استفاده قرار می‌گیرد. یکی از مزایای اصلی مدل ARIMA، سهولت در اجرای آن است. برای اجرای این مدل، تنها نیاز به تعیین سه پارامتر اصلی (p، d و q) می‌باشد. این پارامترها به ترتیب نشان‌دهنده مرتبه خودهمبستگی، مرتبه تفاضل و مرتبه میانگین متحرک هستند.

برای اینکه مدل ARIMA به خوبی عمل کند، پیش‌نیاز مهمی وجود دارد و آن ایستایی سری زمانی است. سری زمانی ایستایی، سری زمانی است که میانگین، واریانس و ساختار همبستگی آن در طول زمان ثابت بماند. به عبارت دیگر، داده‌ها باید حول یک مقدار میانگین مشخص نوسان کنند و واریانس آن‌ها ثابت باشد. در بسیاری از موارد، سری‌های زمانی اولیه ایستایی ندارند. برای رفع این مشکل، از روش تفاضل استفاده می‌شود. در این روش، به جای استفاده از مقدار اصلی سری زمانی، از تفاوت بین مقادیر متوالی آن استفاده می‌شود. به طور ریاضی، اگر t زمان و y(t) مقدار متغیر وابسته در زمان t باشد، تفاضل مرتبه اول به صورت زیر تعریف می‌شود:

Δy(t) = y(t) - y(t-1)

با انجام چندین مرحله تفاضل (در صورت نیاز)، می‌توان به یک سری زمانی ایستایی دست یافت.

پارامتر **q**، که به عنوان جزء میانگین متحرک شناخته می‌شود، بخش نهایی مدل ARIMA است. این پارامتر تعداد خطاهای گذشته‌ای را نشان می‌دهد که بر مقدار فعلی تأثیر می‌گذارند. مفهوم اصلی مدل میانگین متحرک (MA) این است که خطاهای گذشته (ωₜ₋ₖ) بر مقادیر آینده سری زمانی تأثیر می‌گذارند.

y(t) = μ + φ₁y(t-1) + φ₂y(t-2) + ωₜ

با ترکیب اجزای AR و MA، مدل ARIMA یک چارچوب انعطاف‌پذیر برای مدل‌سازی همبستگی و میانگین متحرک در داده‌های سری زمانی فراهم می‌کند.

* **y(t):** مقدار پیش‌بینی شده سری زمانی در زمان t است.
* **μ:** میانگین سری زمانی است (به دلیل ایستایی، این مقدار ثابت است).
* **φ₁**, **φ₂:** ضرایب خودهمبستگی هستند که تأثیر مقادیر گذشته بر مقدار فعلی را نشان می‌دهند.
* **ωₜ:** یک خطای تصادفی است که معمولاً فرض می‌شود توزیع نرمال دارد و تغییرات نامشخص را توضیح می‌دهد.
* **p:** مرتبه جزء خودهمبستگی است و تعداد دوره‌های زمانی گذشته‌ای را نشان می‌دهد که در مدل لحاظ می‌شود.

پارامتر **q**، که به عنوان جزء میانگین متحرک شناخته می‌شود، بخش نهایی مدل ARIMA است. این پارامتر تعداد خطاهای گذشته‌ای را نشان می‌دهد که بر مقدار فعلی تأثیر می‌گذارند. مفهوم اصلی مدل میانگین متحرک (MA) این است که خطاهای گذشته (ωₜ₋ₖ) بر مقادیر آینده سری زمانی تأثیر می‌گذارند.

با ترکیب اجزای AR و MA، مدل ARIMA یک چارچوب انعطاف‌پذیر برای مدل‌سازی همبستگی و میانگین متحرک در داده‌های سری زمانی فراهم می‌کند.

y(t) = μ + ωₜ + θ₁ωₜ₋₁ + θ₂ωₜ₋₂

مدل ARIMA(0,0,2) بر اساس خطاهای پیش‌بینی در دو دوره زمانی قبلی (t-1 و t-2) عمل می‌کند. ضرایب θ نشان می‌دهند که این خطاهای گذشته تا چه اندازه بر مقدار فعلی سری زمانی تأثیرگذار هستند. به عبارت دیگر، اگر در پیش‌بینی‌های قبلی خطای بزرگی داشته باشیم، این خطا در پیش‌بینی فعلی نیز تأثیرگذار خواهد بود.

کتابخانه statsmodels در پایتون به طور خودکار بهترین مقادیر برای پارامترهای p، q و d را پیدا می‌کند. این پارامترها مشخص می‌کنند که مدل ما تا چه اندازه به مقادیر گذشته و خطاهای گذشته وابسته است. با استفاده از این پارامترها، می‌توانیم مدل ARIMA را روی داده‌های آموزشی برازش دهیم و سپس از آن برای پیش‌بینی مقادیر آینده استفاده کنیم. توابع fit() و predict() در پایتون برای این کار مورد استفاده قرار می‌گیرند.

**2.1 مدل میانگین متحرک خودهمبسته تلفیقی (ARIMA)**

مدل میانگین متحرک خودهمبسته تلفیقی (**ARIMA**) ترکیبی از فرآیندهای خودهمبسته (**AR**) و میانگین متحرک (**MA**) همراه با تفاضل‌گیری است که برای تحلیل داده‌های سری زمانی غیرایستا استفاده می‌شود. تفاضل‌گیری می‌تواند برای داده‌های غیر فصلی (مرتبه «d») و داده‌های فصلی (مرتبه «D») اعمال شود. بسته به وجود یا عدم وجود فصلی بودن، مدل ARIMA به دو دسته **فصلی** و **غیرفصلی** تقسیم می‌شود. برای مدل فصلی، یک مؤلفه اضافی برای نمایش فصلی بودن اضافه می‌شود.

**مدل ARIMA غیرفصلی**

فرم کلی مدل ARIMA غیرفصلی که با **ARIMA(p,d,q)** نمایش داده می‌شود، به صورت ریاضی به شکل زیر است:

که در آن:

* : نشان‌دهنده مرتبه خودهمبستگی ()، مرتبه تفاضل‌گیری () و مرتبه میانگین متحرک () است.
* : ضرایب مدل خودهمبسته غیرفصلی با مرتبه :
* : ضرایب مدل میانگین متحرک غیرفصلی با مرتبه :
* : باقیمانده‌ها (مقادیر خطای تصادفی) در زمان .
* : میانگین فرآیند را نشان می‌دهد اگر باشد، یا ضریب روند قطعی اگر .

**مدل ARIMA فصلی**

برای داده‌های فصلی، مدل **ARIMA فصلی** به صورت **ARIMA(P,D,Q)ₛ** نمایش داده می‌شود، که یک مؤلفه فصلی اضافی در آن لحاظ شده است. مدل ضربی باکس-جنکینز برای ARIMA فصلی به صورت زیر بیان می‌شود:

که در آن:

* : نشان‌دهنده مرتبه خودهمبستگی فصلی ()، مرتبه تفاضل‌گیری فصلی () و مرتبه میانگین متحرک فصلی () است.
* : نشان‌دهنده دوره فصلی است.
* : ضرایب مدل خودهمبسته فصلی با مرتبه :
* : ضرایب مدل میانگین متحرک فصلی با مرتبه :

**2.2 مدل ARIMA با متغیرهای برون‌زا (ARIMAX)**

مدل **ARIMAX** نسخه توسعه‌یافته ARIMA است که متغیرهای برون‌زا را نیز در بر می‌گیرد. این متغیرها عوامل خارجی هستند که می‌توانند بر سری زمانی مورد بررسی تأثیر بگذارند. به عنوان مثال، در مطالعات مرتبط با روندهای اقتصادی، متغیرهای خارجی مانند دوره‌های انتخاباتی یا تغییرات سیاستی ممکن است در مدل لحاظ شوند.

**مدل ARIMAX فصلی**

فرم کلی مدل ARIMAX فصلی به صورت زیر است:

که در آن:

* : متغیرهای برون‌زا در زمان .
* : ضرایب متغیرهای برون‌زا.

**انواع کلیدی:**

1. **ARIMAX با روند تصادفی**: این مدل شامل تفاضل‌گیری () برای مدیریت غیرایستایی است:
2. **ARIMAX با روند قطعی**: این مدل فرض می‌کند که روند قطعی بدون نیاز به تفاضل‌گیری وجود دارد:

در اینجا، نشان‌دهنده متغیرهای ساختگی برای رویدادهای خاص یا تغییرات تقویمی است و ضریب روند قطعی می‌باشد.

**Result**

A graph showing a price prediction

Description automatically generated with medium confidence

A graph showing a price

Description automatically generated

A graph showing a line graph

Description automatically generated with medium confidence

Table 1. Evaluating the prediction error of models

|  |  |  |  |
| --- | --- | --- | --- |
|  | ARIMAX | LSTM | Enhanced GRU |
| MSE | 0.0001 | 0.0068 | 0.9839 |
| MAE | 0.00009 | 0.00685 | 0.98449 |
| R-Squared | 0.0000963 | 0.0069965 | 0.98402 |

**References**

* <http://applsci-13-08813-v2.pdf>
* <http://Text-39747-1-10-20240321.pdf>
* <https://ptihiup2018.blogspot.com/2019/07/artifical-neural-network.html>