پیش بینی قیمت سهام بازار بورس ایران با استفاده از روش LSTM

استاد مربوطه: دکتر منصور رزقی

پژوهشگر: ساجده لشگری

مرداد ۹۹

دانشگاه تربیت مدرس - گروه علم داده‌ها

فهرست مطالب

1. مقدمه

پیش بینی قیمت سهام نقش مهمی در تعیین استراتژی معاملات یا تعیین زمان مناسب برای خرید یا فروش سهام را دارا می‌باشد.

در این پروژه هدف پیش بینی قیمت و روند بازار سهام بورس است. یکی از روش‌های سرمایه گذاری، سرمایه گذاری در بورس می‌باشد که با ساخت مدل مناسب برای پیش بینی روند قیمت سهام خریداری شده یا در نظر گرفته شده، می‌توان به موقع به خرید و فروش پرداخت و سود حاصل از سرمایه گذاری را افزایش داد.

پیش بینی بازار سهام یکی از دشوارترین کارها در زمینه محاسبات و پیش بینی می‌باشد و عوامل زیادی بر حرکت قیمت سهام مانند نرخ تورم، محیط اقتصادی، مسائل سیاسی، رفتارهای عقلانی یا غیرمنطقی، احساس سرمایه گذار، شایعات بازار، مدیریت ضعیف شرکت و غیره تأثیر می‌گذارد و همه این جنبه‌ها باعث می‌شود قیمت سهام بی‌ثبات، نوسانات شدید، پیچیدگی زیاد و در نتیجه پیش بینی آن بسیار دشوار باشد.

این بدان معنی است که هیچ الگوی ثابتی در داده‌ها وجود ندارد که به شما امکان دهد قیمت سهام را با گذشت زمان، دقیقاً مدل کنید و همانطور که برتن ملکیل (Burton Malkiel) اقتصاددان دانشگاه پرینستون، در کتاب خود با عنوان "A Random Walk Down Wall Street" می‌گوید: اگر بازار واقعاً کارآمد باشد و قیمت سهم تمام عوامل را سریعاً منعکس کند با یک فرآیند تصادفی رو به رو هستیم و هیچ امیدی برای یادگیری ماشین وجود ندارد.

در واقع در این موضوع ما تصمیم داریم که از داده‌ها الگوبرداری کنیم، به صورتی که پیش بینی‌های انجام شده با رفتار واقعی داده‌ها ارتباط داشته باشد به عبارت دیگر، هدف و نیاز ما دستیابی به مقادیر دقیق سهام در آینده نیست بلکه روند و حرکات قیمت سهام (یعنی نزولی یا صعودی بودن آن) در آینده مورد توجه می‌باشد.

به طور کلی برای تحلیل و پیش بینی بازار سهام سه روش اصلی وجود دارد:

۱) تجزیه و تحلیل بنیادی (Fundamental analysis)
۲) تجزیه و تحلیل تکنیکی و فنی (Technical analysis)
۳) تجزیه و تحلیل سری زمانی (Time series analysis)

پیش از استفاده از شبکه‌های عصبی برای پیش بینی قیمت سهام، بسیاری از افراد از روش‌های تحلیل تکنیکی و فنی برای ارزیابی وضعیت سهام برای خرید و فروش و نگهداری استفاده می‌کردند که در این پروژه روش سوم مورد توجه و استفاده قرار گرفته است.

روش‌های پیش بینی با استفاده از تحلیل سری زمانی، دارای دو بخش خطی و غیر خطی می‌باشد.

مدل‌های خطی مانند AR، MA، ARIMA، ARMA و CARIMA هستند که همیشه عملکرد خوبی ندارند و محدودیت‌های خاصی برای تحقق فرضیات آماری در آن‌ها وجود دارد، مانند فرض نرمال بودن.

برای بهبود این مدل‌ها نیاز به داده‌های گذشته بیشتری وجود دارد، همچنین این روش‌ها فقط روابط خطی بین داده‌ها را در نظر می‌گیرند. برای رفع این مشکل مدل‌های غیر خطی مانند ARCH، GARCH، ANN، RNN، LSTM و ... به وجود آمده اند و طبق ادعای یکی از مقالات درج شده در بخش منبع، تکنیک‌های زیادی بر روی مجموعه‌هایی از داده‌های مالی اجرا شده اند و نتایج نشان داده است که LSTM به مراتب برتر از ARIMA عمل می‌کند.

همچنین از آنجایی که بورس سهام به عنوان سیستم‌های دینامیکی غیرخطی و غیر پارامتری در نظر گرفته می‌شوند، برای بهبود عملکرد پیش بینی، روش‌های انعطاف پذیرتری که بتوانند ابعاد پیچیده‌تری را یاد گیرند، ضروری است.

به طور سنتی، برای پیش بینی قیمت آینده سهام بورس از روش‌های استراتژی کمی مانند رگرسیون خطی، مدل ARIMA و همچنین مدل GARCH استفاده می‌شود که با پیشرفت و توسعه علم در این زمینه، با توجه به مقاله‌های بررسی شده ثابت شد که این روش‌ها برای مدت معینی در گذشته موثر بوده و با تغییرات به وجود آمده در صنعت مالی، این مدل‌ها کمتر اثربخش شدند، بنابراین صنعت تجارتِ کمی، به روش‌های یادگیری عمیق روی آورده و امروزه بیشتر روش LSTM مورد استفاده قرار می‌گیرد.

هدف اصلی این پروژه نیز ساختن یک مدل برای پیش بینی با استفاده از مدل LSTM می‌باشد که در بخش۴ توضیح داده می‌شود.

برای ایجاد مدل پیش بینی کننده قیمت سهام، چهار گامِ **جمع آوری داده‌ها** و **پیش پردازش** آن‌ها، **مدلسازی و پیش بینی** آن و **ارزیابی نتایج تجربی** باید طی شود که در بخش‌های بعد مفصل مورد بررسی قرار می‌گیرد.

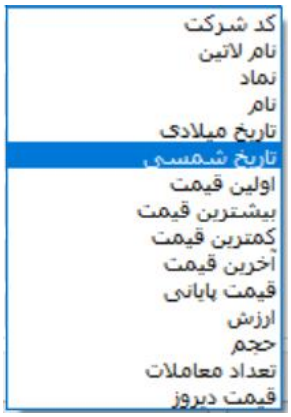
2. مجموعه داده‌ها

مجموعه داده‌هایی که در این پروژه در نظر گرفته شده مجموعه داده‌های سهام بزرگ **اخابر** است که مربوط به شرکت مخابرات ایران می‌باشد.

2.1. جمع آوری داده‌ها

این داده‌ها با استفاده از نرم افزار Tse client در سایت **شرکت مدیریت فناوری بورس تهران** جمع آوری شده و در مجموع همانطور که در شکل۱ قابل مشاهده است، شامل ۱۵ متغیر می‌باشد و در بین آن‌ها ۸ متغیر تاریخ شمسی (Date-S) و میلادی (Date-G)، اولین قیمت (Open: قیمت باز شده سهام در روز)، بیشترین (Highest) و کمترین قیمت (Lowest)، آخرین قیمت (Last: قیمت بسته شده سهام در روز)، قیمت پایانی (Close: میانگین قیمت سهام در روز) و حجم معاملات (Volume) با توجه به نیاز و هدف مسئله، از سایت استخراج شده و در فایل اکسل قرار داده شده است.

اطلاعات جمع آوری شده تا تاریخ ۲۵ تیر ماه ۱۳۹۹ (به میلادی ۲۰۲۰/۰۷/۱۵) را شامل می‌شود.



شکل ۱

2.2. توصیف داده‌ها

مجموعه داده‌های جمع آوری شده سهام اخابر شامل ۸ متغیر (توضیح داده شده در بخش ۲,۱.) و ۲۸۷۷ مشاهده می‌باشد که شامل ۱۲ سال یعنی از تاریخ ۱۳۸۷/۰۵/۱۹ تا ۱۳۹۹/۰۴/۲۵ است.

اطلاعات کاملتری از متغیرها و داده‌ها در جدول ۱ و شکل ۲ مشاهده می‌شود.

نام متغیر	Date-S	Date-G	Open	Highest	Lowest	Last	Close	Volume
توصیفی از آن‌ها	تاریخ شمسی	تاریخ میلادی	اولین قیمت	بیشترین قیمت	کمترین قیمت	آخرین قیمت	قیمت پایانی	حجم معاملات

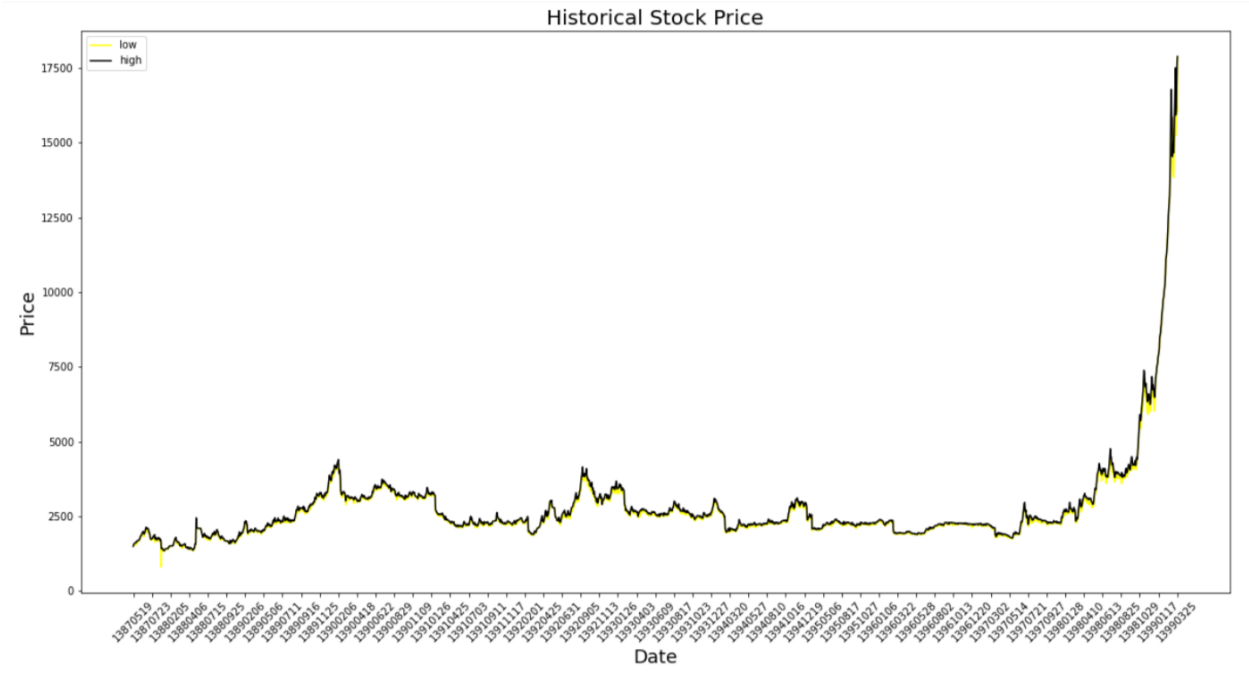
جدول ۱: خلاصه‌ای از متغیرهای مورد مطالعه

	Date-S	Date-G	Open	Close	Highest	Lowest	Last	Volume
count	2.522000e+03	2.522000e+03	2522.000000	2522.000000	2522.000000	2522.000000	2522.000000	2.522000e+03
mean	1.393056e+07	2.014267e+07	2771.473037	2772.749009	2805.255749	2733.337827	2767.065424	1.516675e+07
std	3.325044e+04	3.347841e+04	1612.502771	1591.238263	1630.482017	1581.888177	1612.236949	1.890308e+08
min	1.387052e+07	2.008081e+07	808.000000	1362.000000	1349.000000	808.000000	1335.000000	3.672000e+03
25%	1.390083e+07	2.011112e+07	2177.500000	2194.000000	2200.000000	2155.000000	2175.000000	1.738965e+06
50%	1.393061e+07	2.014087e+07	2349.000000	2352.000000	2373.500000	2315.000000	2341.500000	4.143740e+06
75%	1.396053e+07	2.017082e+07	2989.250000	2999.250000	3040.000000	2929.500000	2995.250000	1.077948e+07
max	1.399033e+07	2.020062e+07	17880.000000	17880.000000	17880.000000	17880.000000	17880.000000	9.174731e+09

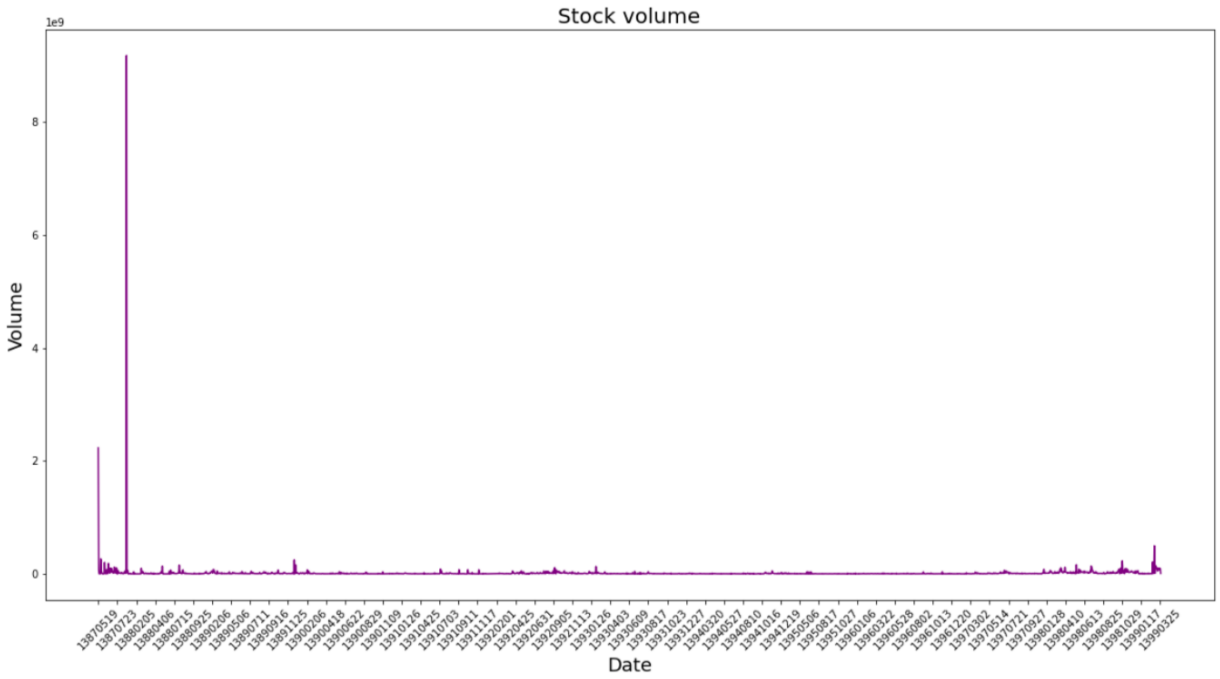
شکل ۲

همچنین در شکل‌های ۳ و ۴و ۵ روند کلی از داده‌ها (قیمت‌های بازشده، بسته شده، نهایی و آخرین قیمت، پایین‌ترین قیمت، بالاترین قیمت و حجم معاملات) نمایش داده شده است.

شکل۳



شکل۴



شکل۵

3. پیش پردازش داده‌ها

در این بخش به پیش پردازش داده‌ها پرداخته می‌شود.

3.1. بررسی و رسیدگی به داده‌های گمشده

ابتدا گمشدگی داده‌ها را بررسی کرده، با توجه به اینکه داده‌ها در روزهای بدون معامله (یعنی روزهایی که سهم بسته است، به دلیل به مجمع رفتن و سود بیش از ۵۰٪ در بازه زمانی کم و مشخصی و مواردی از این قبیل) نیز جمع آوری شده اند باعث به وجود آمدن داده‌های گمشده شده در مجموعه داده‌های در حال بررسی می‌باشد که با فیلتر گذاری در مرحله جمع آوری و ذخیره سازی ثانویه داده‌ها این مشکل رفع شده است.

با اعمال فیلتر روی روزهای با معامله، داده‌های بدست آمده دارای همان شرایط هستند فقط اندازه مشاهدات سهام اخابر به ۲۵۲۲ تغییر کرده و در این صورت آخرین داده مربوط به تاریخ ۱۳۹۹،۰۳،۲۶ می‌باشد و در این حالت، همانطور که در شکل۶ نمایش داده شده، داده‌ها همگی مشاهده شده (recorded) می‌باشند و داده گمشده ای وجود ندارد.

```
Date-S     0
Date-G     0
Open       0
Close      0
Highest    0
Lowest     0
Last       0
Volume     0
dtype: int64
```

شکل۶

3.2. مرتب سازی داده‌ها

در گام بعد، از مرتب بودن داده‌ها بر اساس زمان آن‌ها مطمئن شده و با انجام عملیات sorted by index of time براساس تاریخ شمسی همه داده‌ها به ترتیب صعودی (از گذشته به حال) مرتب می‌شوند.

(لازم به ذکر است که اشاره شود داده‌ها به صورت مرتب جمع آوری شده و این کار صرفاً جهت اطمینان بیشتر و همچنین برای بیان مراحل به صورت کامل انجام شده، از همین‌رو شکل‌های۳، ۴ و ۵ به درستی در مرحله قبل رسم شده اند.)

شمای کلی داده‌ها در شکل۷ مشاهده می‌شود.

	Date-S	Date-G	Open	Close	Highest	Lowest	Last	Volume
0	13870519	20080809	1500	1500	1500	1500	1500	2230300000
1	13870520	20080810	1545	1545	1545	1545	1545	64168880
2	13870521	20080811	1591	1553	1591	1553	1553	3380431
3	13870522	20080812	1599	1556	1599	1556	1556	1062213
4	13870523	20080813	1602	1563	1602	1563	1563	2628987
...	...	...	...	...	...	...	...	...
2517	13990320	20200609	15300	15375	15920	15240	15499	103601781
2518	13990321	20200610	15998	16118	16143	15623	16143	85739555
2519	13990324	20200613	16750	16850	16920	16250	16920	79411198
2520	13990325	20200614	17200	17030	17690	16010	17690	95462135
2521	13990326	20200615	17880	17880	17880	17880	17880	8395022

2522 rows × 8 columns

شکل۷

3.3. تقسیم بندی داده‌ها به داده‌های آموزش و آزمایش

ابتدا داده‌ها را بر اساس تاریخ شمسی ایندکس گذاری کرده و از تاریخ ۱۳۹۸,۰۳,۰۲۶ تا ۱۳۹۹,۰۳,۰۲۶ یعنی یک سال آخر را به عنوان داده‌های آزمایش و مابقی را به عنوان داده‌های آموزش در نظر گرفته، به عبارتی همانطور که در شکل۸ و ۹ مشاهده می‌شود ۲۳۷ تا از داده‌ها یعنی حدوداً ۱۰٪ کل داده‌ها، داده‌های آزمایش در نظر گرفته شده اند.

Date-S	Date-G	Open	Close	Highest	Lowest	Last	Volume
13980326	20190616	2900	3050	3072	2880	3000	95174024
13980327	20190617	2930	2956	3070	2898	2898	45460705
13980328	20190618	2861	2820	2934	2809	2811	46334129
13980329	20190619	2805	2756	2820	2699	2766	22214608
13980401	20190622	2700	2718	2820	2650	2729	20073055
...	...	...	...	...	...	...	...
13990320	20200609	15300	15375	15920	15240	15499	103601781
13990321	20200610	15998	16118	16143	15623	16143	85739555
13990324	20200613	16750	16850	16920	16250	16920	79411198
13990325	20200614	17200	17030	17690	16010	17690	95462135
13990326	20200615	17880	17880	17880	17880	17880	8395022

237 rows × 7 columns

شکل۸: نمایشی از داده‌های آزمایش

Date-S	Date-G	Open	Close	Highest	Lowest	Last	Volume
13870519	20080809	1500	1500	1500	1500	1500	2230300000
13870520	20080810	1545	1545	1545	1545	1545	64168880
13870521	20080811	1591	1553	1591	1553	1553	3380431
13870522	20080812	1599	1556	1599	1556	1556	1062213
13870523	20080813	1602	1563	1602	1563	1563	2628987
...	...	...	...	...	...	...	...
13980319	20190609	2400	2397	2424	2376	2420	26049703
13980320	20190610	2430	2491	2516	2424	2516	31570163
13980321	20190611	2615	2678	2745	2615	2739	77803753
13980322	20190612	2749	2806	2811	2741	2811	41143361
13980325	20190615	2848	2926	2946	2801	2946	107615988

2285 rows × 7 columns

شکل۹: نمایشی از داده‌های آموزش

3.4. نرمال سازی داده‌ها

این مرحله به نرمال سازی داده‌ها با روش Min Max Scaling پرداخته و همه داده‌ها را به اعداد بین ۰ تا ۱ تبدیل کرده تا مدلسازی داده‌ها (به خصوص زمانی که از تابع فعالسازی سیگموید استفاده می‌شود) بهتر صورت گیرد.

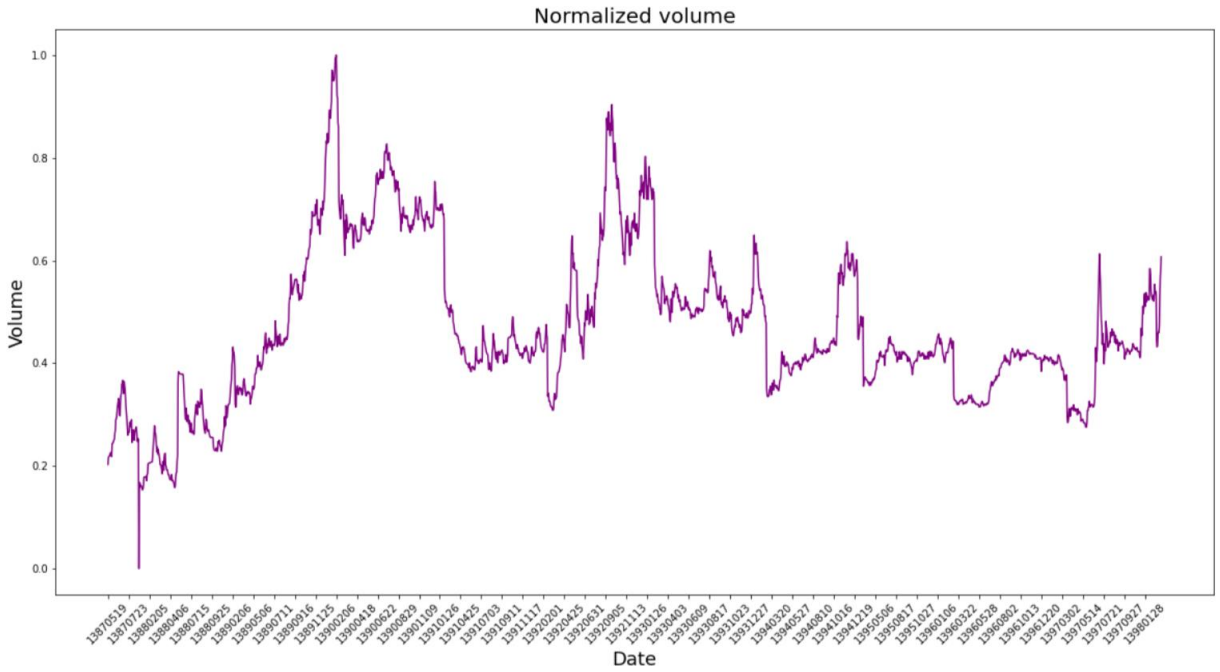
در شکل‌های زیر شمای کلی داده‌های نرمالسازی شده در قالب جدول و نمودار نمایش داده شده است.

(در شکل‌های ۱۰، ۱۱، ۱۲ نرمالسازی روی داده‌های آموزش نشان داده شده است)

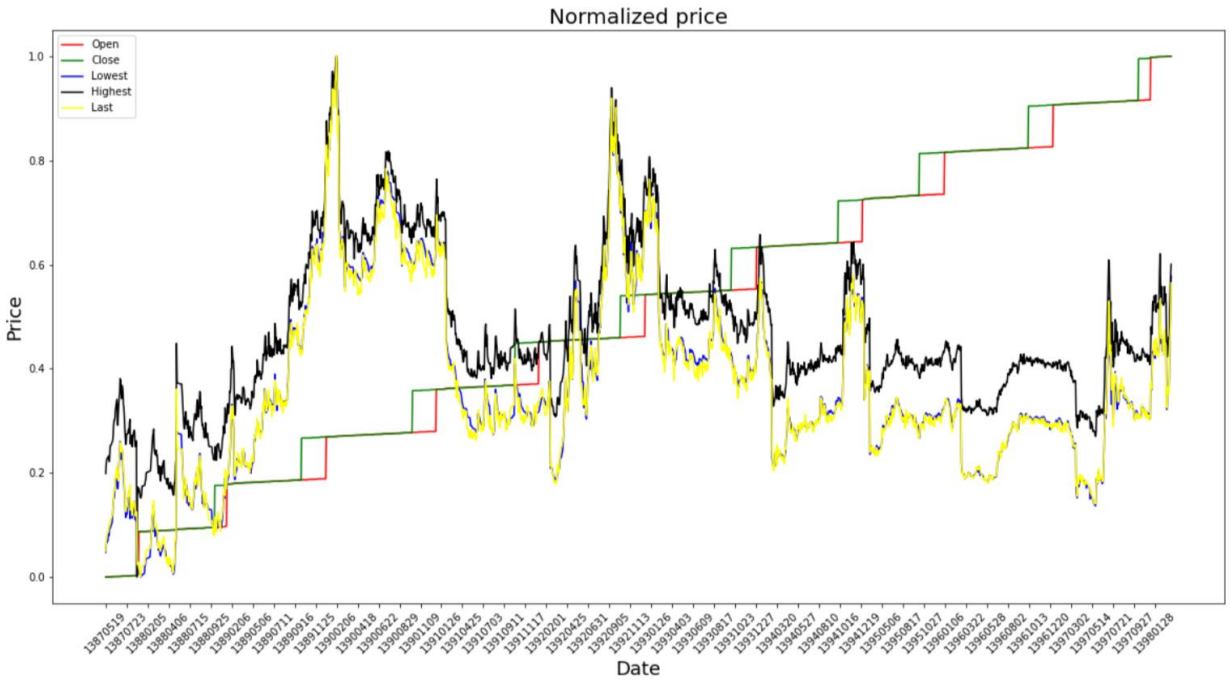
	Date-S	Date-G	Open	Close	Highest	Lowest	Last	Volume
0	13870519	20080809	0.000000	0.000000	0.198679	0.047244	0.049492	0.202814
1	13870520	20080810	0.000009	0.000009	0.211599	0.062650	0.064241	0.216002
2	13870521	20080811	0.000018	0.000018	0.224806	0.065389	0.079318	0.218347
3	13870522	20080812	0.000027	0.000027	0.227103	0.066416	0.081940	0.219226
4	13870523	20080813	0.000036	0.000036	0.227964	0.068812	0.082924	0.221278
...	...	...	...	...	...	...	...	...
2281	13980320	20190610	0.999945	0.999945	0.465690	0.386511	0.382498	0.473623
2282	13980321	20190611	0.999954	0.999954	0.518806	0.450531	0.457555	0.529601
2283	13980322	20190612	0.999964	0.999964	0.557278	0.494351	0.479187	0.566530
2284	13980325	20190615	0.999991	0.999991	0.585702	0.535433	0.523435	0.584115
2285	13980326	20190616	1.000000	1.000000	0.600632	0.577884	0.564733	0.607268

2286 rows × 8 columns

شکل ۱۰



شکل ۱۱



شکل۱۲

3.5. کاهش ابعاد

با توجه به مسئله و شبکه طراحی شده، هدف پیش بینی قیمت پایانی سهم (Close) می‌باشد. در این مرحله به کاهش بعد پرداخته، در واقع انتخاب ویژگی (feature selection) صورت می‌گیرد پس متغیر مورد استفاده به صورت شکل۱۳ به دست می‌آید.

	Close
0	0.000000
1	0.000009
2	0.000018
3	0.000027
4	0.000036
...	...
2281	0.999945
2282	0.999954
2283	0.999964
2284	0.999991
2285	1.000000

2286 rows × 1 columns

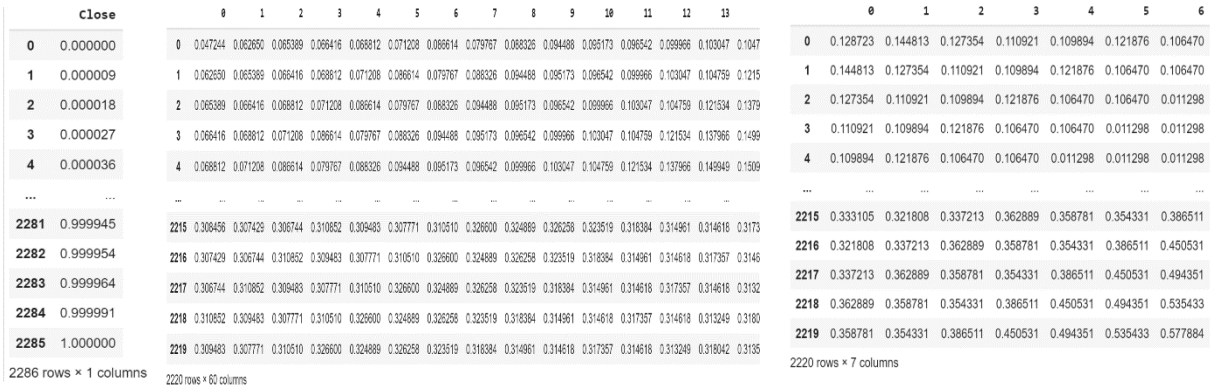
شکل۱۳

می‌توان بار دیگر به جای انتخاب قیمت پایانی با متغیرهای دیگر شبکه را آموزش داد و در نهایت برای آن‌ها پیش بینی صورت گیرد و یا از شبکه‌های چند متغیره (multi variate) که در بخش۴ معرفی می‌شود، استفاده کرد تا نتایج کامل‌تر و دقیق‌تر بدست آید که در این پروژه به دلیل محدودیت زمانی فرصت پرداختن به آن به صورت عملی و آزمون و خطا نبوده و فقط به عنوان یک پیشنهاد برای ادامه دادن مسیر در این راه، بیان شده است.

3.6. تقسیم بندی داده‌ها به ورودی و خروجی و تنظیم بعد داده‌ها برای شبکه LSTM

در پیش بینی سری زمانی در شبکه LSTM یک رابطه چند به یک وجود دارد. در واقع داده‌ها را با توجه به یک پنجره‌ای بخش بندی می‌کنند که این پنجره در اینجا ۶۰ (window=60) در نظر گرفته شده، از آنجایی که داده‌ها به صورت روزانه جمع آوری شده اند پس معنای این عبارت این می‌باشد که هر بار از داده‌های دو ماه گذشته (۶۰ روز گذشته) استفاده می‌کند و پیش بینی را برای ۷ گام جلوتر انجام می‌دهد یعنی بردارهای ورودی بردارهای ۶۰ تایی می‌باشند که تعدادشان باتوجه به تعداد داده‌ها ۲۲۲۰ است.

به عبارتی همانطور که در شکل۱۴ مشخص شده، ابتدا از داده‌های اول شروع کرده و ۶۰ تای اول داده‌ها را برداشته و داخل ردیف اول قرار داده می‌شوند سپس از داده‌ی بعدی یعنی داده دوم شروع کرده تا به ۶۰ داده رسیده و آن‌ها را در ردیف دوم قرار می‌دهد و به همین ترتیب داده‌های ورودی یعنی x با ابعاد (۶۰ و ۲۲۲۰) برای پیش بینی خروجی یعنی y با ابعاد (۷ و ۲۲۲۰) ساخته شده، حال باید بررسی شود که y به چه صورت به دست می‌آید.



	Close
0	0.000000
1	0.000009
2	0.000018
3	0.000027
4	0.000036
...	...
2281	0.999945
2282	0.999954
2283	0.999964
2284	0.999991
2285	1.000000

2286 rows × 1 columns

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0.047244	0.062650	0.065389	0.066416	0.068812	0.071208	0.086614	0.079767	0.088326	0.094488	0.095173	0.096542	0.099966	0.103047
1	0.062650	0.065389	0.066416	0.068812	0.071208	0.086614	0.079767	0.088326	0.094488	0.095173	0.096542	0.099966	0.103047	0.104759
2	0.065389	0.066416	0.068812	0.071208	0.086614	0.079767	0.088326	0.094488	0.095173	0.096542	0.099966	0.103047	0.104759	0.121534
3	0.066416	0.068812	0.071208	0.086614	0.079767	0.088326	0.094488	0.095173	0.096542	0.099966	0.103047	0.104759	0.121534	0.137966
4	0.068812	0.071208	0.086614	0.079767	0.088326	0.094488	0.095173	0.096542	0.099966	0.103047	0.104759	0.121534	0.137966	0.149949
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2215	0.308456	0.307429	0.306744	0.310852	0.309483	0.307771	0.310510	0.326600	0.324889	0.326258	0.323519	0.318384	0.314961	0.314618
2216	0.307429	0.306744	0.310852	0.309483	0.307771	0.310510	0.326600	0.324889	0.326258	0.323519	0.318384	0.314961	0.314618	0.317357
2217	0.306744	0.310852	0.309483	0.307771	0.310510	0.326600	0.324889	0.326258	0.323519	0.318384	0.314961	0.314618	0.317357	0.314618
2218	0.310852	0.309483	0.307771	0.310510	0.326600	0.324889	0.326258	0.323519	0.318384	0.314961	0.314618	0.317357	0.314618	0.313249
2219	0.309483	0.307771	0.310510	0.326600	0.324889	0.326258	0.323519	0.318384	0.314961	0.314618	0.317357	0.314618	0.313249	0.318042

2220 rows × 60 columns

	0	1	2	3	4	5	6
0	0.128723	0.144813	0.127354	0.110921	0.109894	0.121876	0.106470
1	0.144813	0.127354	0.110921	0.109894	0.121876	0.106470	0.106470
2	0.127354	0.110921	0.109894	0.121876	0.106470	0.106470	0.011298
3	0.110921	0.109894	0.121876	0.106470	0.106470	0.011298	0.011298
4	0.109894	0.121876	0.106470	0.106470	0.011298	0.011298	0.011298
...	...	...	...	...	...	...	...
2215	0.333105	0.321808	0.337213	0.362889	0.358781	0.354331	0.386511
2216	0.321808	0.337213	0.362889	0.358781	0.354331	0.386511	0.450531
2217	0.337213	0.362889	0.358781	0.354331	0.386511	0.450531	0.494351
2218	0.362889	0.358781	0.354331	0.386511	0.450531	0.494351	0.535433
2219	0.358781	0.354331	0.386511	0.450531	0.494351	0.535433	0.577884

2220 rows × 7 columns

شکل۱۴: به ترتیب از سمت چپ به راست مربوط به قسمتی از کل مجموعه داده‌های Close، داده‌های ورودی (x) و داده‌های خروجی (y)

داده‌های خروجی با استفاده از ۶۰ داده ورودی ساخته می‌شوند یعنی ۶۰ داده اول تا ۶۷ امین داده را پیش بینی می‌کنند به این دلیل که طول گام و تاخیر را ۷ در نظر گرفته تا بتوانیم قیمت ۷ روز آینده را در نهایت پیش بینی کنیم.

به این ترتیب داده‌های خروجی ساخته می‌شوند، یک مثال برای فهم راحت تر موضوع: اولین داده خروجی شامل ۷ عدد است که با ۶۰ داده قبل از خود در کل مجموعه داده‌های ما ساخته می‌شود، در واقع داده پیش بینی شده در اینجا مربوط به ۶۱ امین تا ۶۷ امین داده در مجموعه داده کلی ما می‌باشد و دومین داده خروجی نیز شامل ۷ عدد است که مربوط به پیش بینی داده‌های ۶۲ ام تا ۶۸ ام داده‌های کلی می‌باشد و به همین ترتیب هر ۶۰ داده ۷ داده بعدی خود را پیش بینی کرده ولی یکی یکی (cell=1) جلو می‌رود.

حال داده‌های ورودی و خروجی شبکه آماده هستند فقط داده‌های ورودی را به ابعاد (۱ و۶۰ و ۲۲۲۰) درآورده پس حالا می‌توان شبکه را ساخت و مقادیر را پیش بینی کرد که در بخش۴ این موضوع مفصلتر بررسی می‌شود.

4. مدلسازی و پیش بینی داده‌ها

4.1. شبکه LSTM

LSTM (Long-Short Term Memory) یک نوع از شبکه عصبی بازگشتی است که می‌تواند توالی‌های طولانی را یاد بگیرد و پیش بینی کند، به صورتی که از وابستگی طولانی مدت جلوگیری شود و نسبت به مدل بازگشتی سخت‌تر دچار از بین رفتن اثر گذشته یا برعکس زیاد شدن آن و غلبه بر حال (به ترتیب منظور vanishing و exaggerate) می‌شود.

همانطور که پیش‌تر بیان شد، هدف از انجام این پروژه پیش بینی قیمت سهام می‌باشد و برای اینکار از شبکه عصبی LSTM استفاده می‌شود که این شبکه قادر به گرفتن اطلاعات از مراحل گذشته و استفاده از آن برای پیش بینی داده‌های آینده می‌باشد.

پیش بینی با استفاده از روش LSTM با وارد کردن ورودی‌ها و خروجی‌هایی که قبلاً وارد پنجره شده، آغاز می‌شود در اینجا اعداد به شکل ۰ یا ۱ تولید می‌شوند که عدد ۰ بدین معنی است که ورودی فراموش می‌شود و در غیر این صورت عدد ۱ نشان دهنده ادامه یافتن وجود ورودی است. در مرحله بعد، در دروازه ورودی، لایه تعیین می‌شود که داده ها به روز می‌شوند و لایه یک مقدار انتخابی جدید را ایجاد می‌کند. خروجی از لایه دروازه ورودی و لایه به حالت سلول ترکیب می‌شود. در مرحله بعد وضعیت سلول‌های قدیمی با وضعیت سلول جدید به روز می‌شوند.

یکی از انواع مدل LSTM در شکل۱۵ قابل مشاهده می‌باشد.

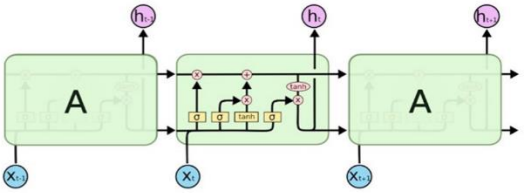


Figure 1. The internal structure of an LSTM [5].

شکل۱۵

همانطور که در شکل۱۵ مشاهده می‌شود معماری در هر نرون به صورتی است که شامل cell، input gate، output gate و forget gate یا همان پنجره می‌باشد که در واقع cell مقادیر را در فواصل زمانی دلخواه به یاد می‌آورد و در اینجا یک در نظر گرفته شده است، input gate همان x ، output gate همان y و forget gate نیز همان پنجره می‌باشد، همچنین سه gate جریان اطلاعات را به داخل و خارج از سلول تنظیم می‌کنند.

اما برخی از انواع مدل LSTM مانند Gated Recurrent (GRU) دروازه خروجی ندارند. شبکه‌های LSTM به طور متداول از داده‌های سری زمانی برای طبقه بندی، پردازش و پیش بینی استفاده می‌شوند.

از مشکلات LSTM ها می‌توان به زمان‌گیر بودن آماده سازی قالب مناسب برای شبکه که هنگام یادگیری به آن نیاز دارد، نام برد. همچنین تعیین اندازه پنجره مناسب و همانند بقیه شبکه‌های عصبی مصنوعی دیگر لایه‌های پنهان، تعداد نرون‌های هر کدام، تابع فعال‌سازی و تابع هدف و مواردی از این قبیل را می‌توان به عنوان چالش‌های آن نام برد.

اگر پنجره زمان خیلی کوچک باشد ممکن است سیگنال‌های قابل توجهی از دست بروند، در حالی که اگر اندازه پنجره زمان خیلی بزرگ باشد، ممکن است اطلاعات نامناسب به عنوان نویز عمل کنند.

با توجه به بررسی پنجره زمانی RNN، بسیاری از مطالعات و رویکردهای کلی مبتنی بر روش‌های آماری یا آزمون و خطا، همراه با اکتشافات مختلفی را پیشنهاد کرده اند. در اینجا با روش آزمون و خطا این مقدار را همانطور که در بخش ۳,۶ بیان شد ۶۰ در نظر گرفته که شکل۱۶ و ۱۷ شاید فهم روند کار، درک ورودی‌ها و خروجی‌های موجود و خروجی‌های مورد انتظار برای پیش بینی را آسان‌تر کند.

شکل۱۶: مربوط به داده‌هایی که در زمان آموزش استفاده می‌شوند که رنگ سبز مربوط به مکانی است که تا آنجا داده ورودی (باتوجه به شکل۱۴ و توضیحات مطرح شده، داده ورودی است X نه مجموعه داده‌های کلی) داریم و ستاره‌ها نماد مقادیر خروجی می‌باشند.

شکل۱۷: سمت راست شکل مربوط به داده‌هایی است که پیش بینی می‌شوند و خط بنفش محل قرار گرفتن آخرین داده‌ای است که داریم و هدف در اینجا پیش بینی داده‌های بین خط بنفش تا خط نارنجی می‌باشد که آن‌ها را الان نداریم و برای آینده هستند، در واقع ستاره‌های نارنجی داده‌های جدیدی هستند که پیش بینی می‌شوند و ستاره‌های آبی خروجی‌هایی هستند که پیش بینی می‌شوند ولی مقدار واقعی آن‌ها را نیز داریم.

4.2. انواع LSTM

4.2.1. مدل‌های Univariate LSTM و Multivariate LSTM

این مدل‌ها به ترتیب هر کدام که یک متغیر و چند متغیر را به عنوان ورودی در نظر می‌گیرد. در اینجا چون یک متغیر "قیمت پایانی سهم" مد نظر می‌باشد پس مدل univariate استفاده می‌شود.

که مدل Mulivariate شامل دو حالت Multiple Input Series و Multiple Parallel Series و همچنین مدل‌های زیر می‌باشد.

4.2.2. Multi-Step LSTM Models و One-Step LSTM Models

در روش One Step طول گام یک می‌باشد و در واقع یک خروجی (گام بعد) پیش بینی می‌شود برای مثال t-1 داده به عنوان ورودی به شبکه داده می‌شود و داده t ام پیش‌بینی می‌شود ولی در روش Multi Step چند خروجی می‌تواند پیش بینی شود و برداری از خروجی‌ها را می‌توان داشت، به عبارتی در این مثال می‌توان علاوه بر یک گام جلو، دو گام جلو و سه گام جلو و ... را پیش بینی کرد.

روش‌های معروفی که در مدل Multi Step استفاده می‌شوند Vector Output Mode و Encoder-Decoder Model می‌باشند.

4.2.3. مدل‌های LSTM

برخی از مدل‌های LSTM در زیر نام برده شده است:

۱. Vanilla LSTM
۲. Stacked LSTM
۳. Bidirectional LSTM
۴. CNN LSTM
۵. ConvLSTM

۱. مدل Vanilla LSTM

در این روش شبکه دارای یک لایه LSTM پنهان و یک لایه خروجی می‌باشد که برای ایجاد یک یا چند پیش بینی (بسته به One Step یا Multi Step بودن هدف) استفاده می‌شود.

۲. مدل Stacked LSTM

مدل Stacked LSTM مانند مدل قبلی است ولی تنها شامل چندین لایه LSTM پنهان می‌باشد.

۳. مدل Bidirectional LSTM

در مورد برخی از مسائل پیش بینی توالی، می‌توان اجازه داد که مدل LSTM توالی ورودی هم رو به جلو و هم رو به عقب باشد و مدل این را بیاموزد، این روش بیشتر برای داده‌های متنی استفاده می‌شود.

۴. مدل CNN LSTM

همانطور که می‌دانیم یک شبکه عصبی پیچشی یا به طور خلاصه CNN نوعی شبکه عصبی است که برای کار با داده‌های تصویر دو بعدی ایجاد شده است.

CNN می‌تواند در استخراج و یادگیری خودکار ویژگی‌های داده توالی یک بعدی مانند داده‌های سری زمانی یک متغیره، بسیار مؤثر باشد. یک مدل CNN را می‌توان در یک مدل ترکیبی با یک پس زمینه LSTM که CNN برای تفسیر پیامدهای ورودی استفاده می‌شود، استفاده کرد که در کنار هم به عنوان دنباله ای از یک مدل LSTM برای تفسیر ارائه می‌شوند. به این مدل ترکیبی CNN-LSTM گفته می‌شود.

اولین قدم تقسیم توالی‌های ورودی به فرعی است که می‌تواند با استفاده از مدل CNN پردازش شود. به عنوان مثال، می‌توان در ابتدا داده‌های سری زمانی تک متغیره خود را به چهار ورودی و خروجی با نمونه ورودی و خروجی تقسیم کرد، سپس هر نمونه می‌تواند به دو نمونه فرعی تقسیم شود، هر یک با دو مرحله زمان. CNN می‌تواند هر پیامد دو مرحله زمانی را تفسیر کند و یک سری زمان تفسیر از پیامدهای مدل LSTM را ارائه دهد تا به عنوان ورودی پردازش شود.

مدل CNN ابتدا یک لایه حلقوی برای خواندن در سراسر پیامد دارد که به تعدادی فیلتر و اندازه هسته نیاز دارد که مشخص شود. تعداد فیلترها تعداد خوانده شده یا تفسیر توالی ورودی است. اندازه هسته تعداد گام‌های زمانی است که شامل هر عمل "خواندن" توالی ورودی است.

۵. مدل ConvLSTM

نوعی LSTM مربوط به CNN-LSTM، ConvLSTM است که در آن خواندن پیچش ورودی مستقیماً در هر واحد LSTM ساخته می‌شود. ConvLSTM برای خواندن داده‌های مکانی و زمانی دو بعدی توسعه داده شد اما می‌تواند برای استفاده با پیش بینی سری زمانی یکپارچه سازگار باشد.

لایه ورودی باید به صورت توالی از تصاویر دو بعدی باشد، بنابراین باید شکل داده‌های ورودی به این صورت باشد:

```
[samples, timesteps, rows, columns, features]
```

در این پروژه از مدل دوم یعنی Stacked LSTM و univariate و multi(7)-Step برای آموزش داده‌ها و در نهایت پیش بینی داده‌ها استفاده می‌شود که در بخش‌های بعدی مفصل‌تر مورد بررسی قرار می‌گیرد.

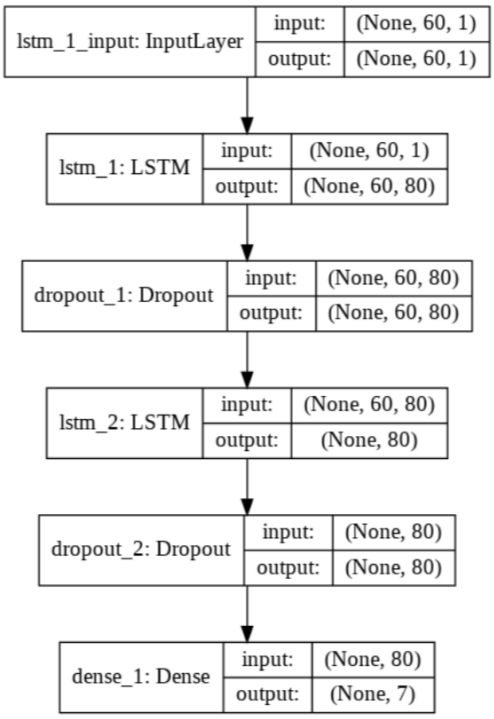
4.3. آموزش شبکه Stacked LSTM روی داده‌ها

داده‌ها، طول گام، پنجره و مدل مورد استفاده پیش‌تر بیان شد و شکل۱۸ خلاصه ای از مدل را نمایش می‌دهد.

همانطور که در شکل نیز مشاهده می‌شود این شبکه دارای دو لایه پنهان با ۸۰ نرون می‌باشد که در بین آن‌ها برای جلوگیری از بیش برازش ۲۰٪ نرون‌ها هر بار در مرحله آموزش روشن و فعال می‌باشند و به این ترتیب تعداد اتصال‌ها و در نتیجه پارامترها کاهش می‌یابد به عبارتی بعد هر لایه پنهان و لایه ورودی از dropout با نرخ ۲۰٪ استفاده شده است و تابع فعال‌سازی مناسب در اینجا با آزمون و خطا relu و تابع هدف mse و تابع بهینه سازی adam در نظر گرفته شده است.

همچنین تعداد نرون‌های لایه خروجی ۷ و تعداد نرون‌های لایه ورودی (۱و ۶۰ و ۲۲۲۰) در نظر گرفته شده و تعداد کل پارامترهای شبکه ۷۸۳۲۷ و تعداد آن‌ها در هر لایه به ترتیب ۲۶۲۴۰، ۵۱۵۲۰ و ۵۶۷ می‌باشد.

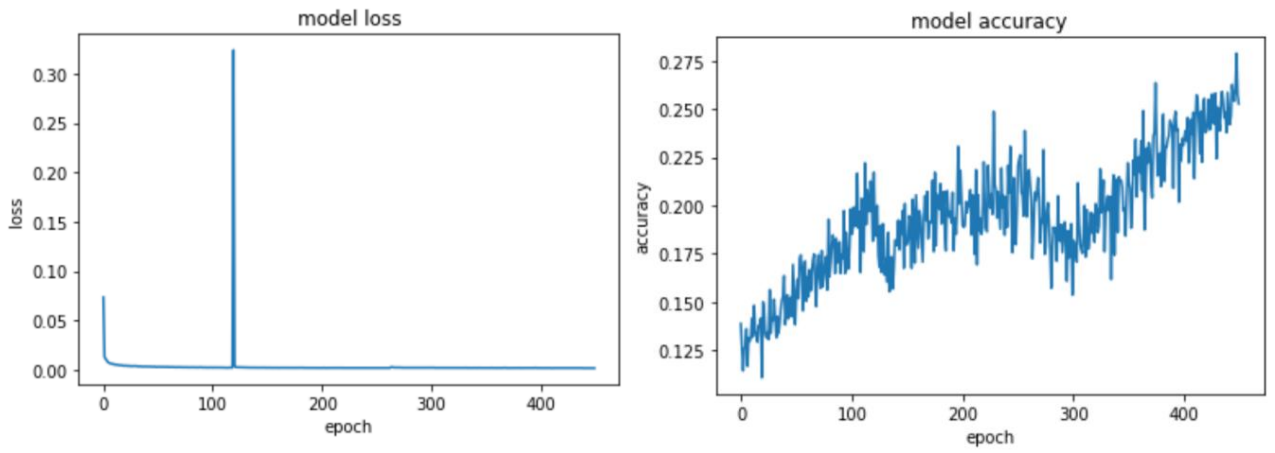
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_1 (LSTM)                (None, 60, 80)            26240
_________________________________________________________________
dropout_1 (Dropout)          (None, 60, 80)            0
_________________________________________________________________
lstm_2 (LSTM)                (None, 80)                51520
_________________________________________________________________
dropout_2 (Dropout)          (None, 80)                0
_________________________________________________________________
dense_1 (Dense)              (None, 7)                 567
=================================================================
Total params: 78,327
Trainable params: 78,327
Non-trainable params: 0
_________________________________________________________________
```



شکل۱۸

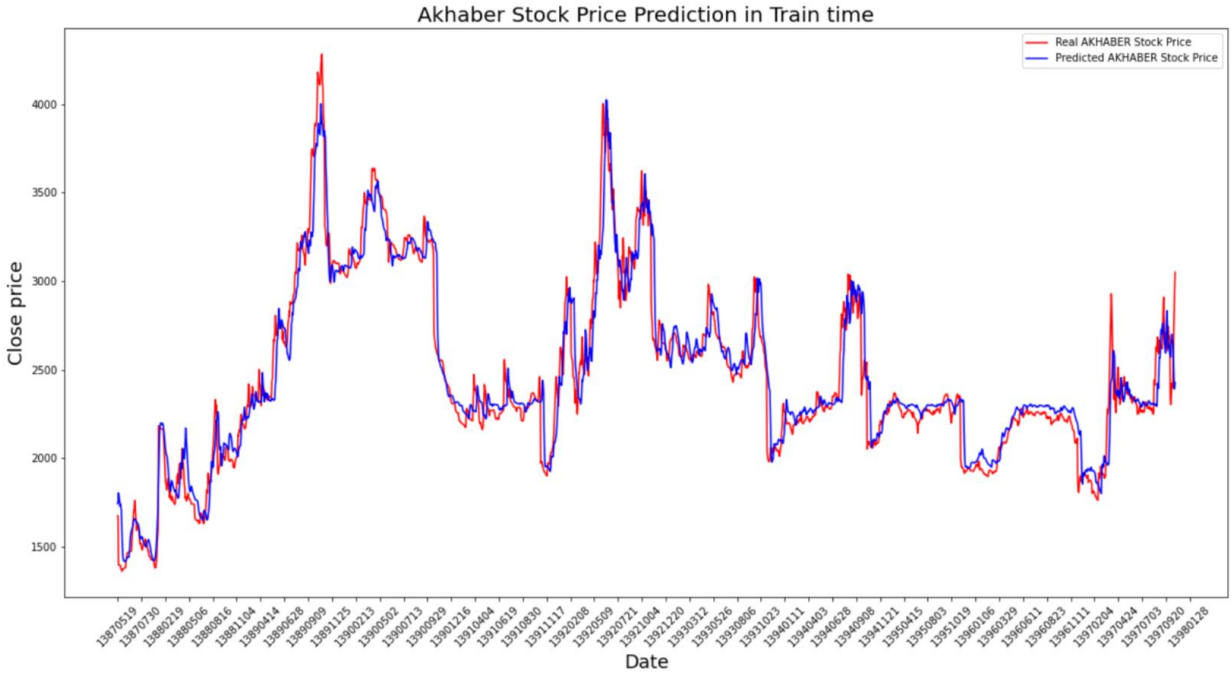
در مرحله بعد مدل را روی داده‌های آموزش برازش داده و مقدار epochs=450, batch_size=64 در نظر گرفته و با توجه به شکل۱۹ به نظر می‌رسد که تقریباً مقادیر epoch و batch مناسب می‌باشند و آموزش به خوبی صورت گرفته است.

(چون انتها نمودار مربوط به epoch صعودی می‌باشد به نظر می‌رسد اگر تعداد بیشتری بهepoch اضافه شود آموزش بهتر صورت می‌گیرد اما با توجه به حجم محاسباتی زیاد و در دسترس نبودن GPU قوی به همین مقدار بسنده شد، از همین رو مقدار دقت خیلی بالا نیست و شاید این مشکل تا حد زیادی با افزایش epoch حل شود).



شکل۱۹: به ترتیب از راست به چپ مربوط به دقت و زیان

همچنین مقادیر پیش بینی شده و واقعی همانطور که در شکل ۲۰ مشاهده می‌شود نزدیک به هم می‌باشند، در نتیجه به نظر می‌رسد که مدل آموزش داده شده به خوبی پیش بینی انجام می‌دهد.



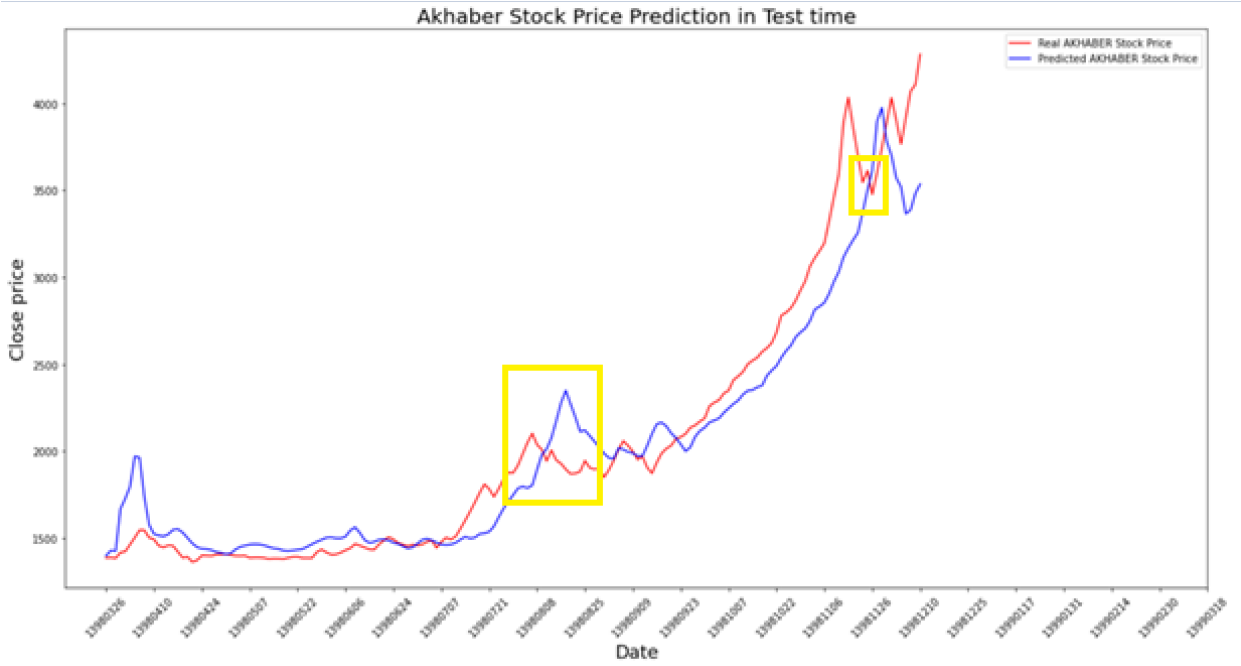
شکل ۲۰

حال برای اطمینان بیشتر مدل را روی داده‌های آزمایش برازش می‌دهیم و شکل۲۱ نتیجه را نمایش داده است، که باتوجه به آن مدل برای این داده‌ها نیز عملکرد تقریباً خوبی داشته است.

به طور کلی همانطور که در بخش مقدمه صحبت شد، در اینجا انتظار نداریم که خود قیمت‌ها به طور دقیق پیش بینی شوند اما روند تقریباً باید دقیق پیش بینی شود، باتوجه به شکل۲۱ در دو قسمتی که با رنگ زرد مشخص شده این موضوع برقرار نیست و روند به درستی پیش‌بینی نشده است که ممکن است به دلیل بیش برازش داده‌ها باشد.

به طور کلی شاید هنوز هم نیاز باشد که با آزمون و خطا در مورد هایپر پارامترهایی همچون تعداد نرون‌ها، تابع فعالسازی و ... تجدید نظر انجام داده و بتوان مدل را بهبود داد که باتوجه به مهلت تعیین شده برای تحویل پروژه، بهترین مدل دست یافته تا به اینجا مدل مطرح شده می‌باشد.

از همین‌رو کار با همین مدل ادامه داده می‌شود.



شکل ۲۱

برای سنجیدن بهتر مدل مقادیر دقت و زیان آزمایش و آموزش را محاسبه کرده که در شکل۲۲ قابل مشاهده می‌باشند.

```
accuracy : 21.64%
loss : 0.45%
accuracy : 24.95%
loss : 0.13%
```

شکل ۲۲

حال می‌توان با قرار دادن ۶۰ داده آخر (۶۰ داده آخر در داده‌های آزمایش) یعنی داده‌های ۲۰ اسفند ۹۸ تا ۲۶ خرداد ۹۹، قیمت سهم را تا یک هفته کاری یعنی ۷ روز بعد از آن که الان نداریمش پیش بینی کنیم.

ابتدا مدل یاد گرفته را ذخیره کرده و بعد همانطور که گفته شد برای پیش بینی داده‌های جدید از آن استفاده می‌شود.

با انجام این کار داده‌های زیر بدست می‌آیند.

```
array([[17673.469, 17667.637, 17604.256, 17617.543, 17551.352, 17536.445,
        17487.555]], dtype=float32)
```

همانطور که مشاهده می‌شود، در روز اولی که سهام باز می‌شود قیمت ۱۷۶۷۴ هزار ریال، بعد از آن ۱۷۶۶۸ و در آخرین روز ۱۷۴۸۸ می‌شود که با توجه به داده‌های بدست آمده می‌توان گفت، در اولین روزی که سهام باز می‌شود قیمت افزایش یافته بعد از آن تا روز سوم روند کاهشی داشته و روز چهارم کمی افزایش یافته و دوباره بعد از آن قیمت نزول می‌کند.

5. پیاده سازی

برای پیاده سازی شبکه توصیف شده کتابخانه‌ها و تنظیمات زیر مورد استفاده قرار گرفته اند.

Epochs =450 , Batch size =64

sudo pip install h5py → برای ذخیره کردن شبکه و پارامترها و ... آن از این کتابخانه استفاده می‌شود

```
from keras.models import Sequential

from keras.layers import Dense

from keras.layers import LSTM

from keras.layers import Dropout

from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from keras.models import load_model
from keras.utils.vis_utils import plot_model
```

6. نتیجه گیری تجربی

در این پروژه تلاش شد با مطالعه مطالبی در مورد LSTM و روش‌های مختلف آن، داده‌های قیمت سهام بورس و پیش بینی کردن آن‌ها و همچنین تاحدودی به مطالب کلی تر از روش LSTM در پیش بینی قیمت سهام بورس، در حد بیان کلیات و دادن دید کلی از روش‌ها در بخش مقدمه و بخش‌های دیگر پرداخته شد که منابع مطالعه شده در بخش منابع، قابل مشاهده می‌باشد.

به طور کلی روش‌های معروف بیان شده در مقالات مختلف برای پیش بینی قیمت سهام در شکل۲۳ به همراه اطلاعات کلی هر یک از روش‌ها نمایش داده شده است.

و در این پروژه تلاش شد از مدل Stacked LSTM برای آموزش الگوی داده‌ها و در نتیجه پیش بینی داده‌ها تا ۷ روز آینده استفاده شود که همانطور که در شکل‌های ۲۰، ۲۱ و ۲۲ مشاهده شد، مدل تقریبا خوبی آموزش دیده شده است اما همانطور که پیش‌تر نیز گفته شد پیش بینی قیمت سهام بورس به دلایل نوسانات زیاد و موارد دیگری که گفته شد خیلی دشوار است و ممکن است نتایج کاملاً قابل اطمینان نباشد از همین رو نظر پژوهشگر این پروژه علی رغم تلاش‌هایی که برای یافتن نتیجه درست و قابل اعتماد حاصل شد و ارزیابی مدل که تقریباً نتیجه خوبی را نشان می‌دهد، این است که تصمیم گیری بر اساس نتایج همچنان پر ریسک می‌باشد.

Table 1. A summary of recent studies on stock market prediction.

Authors (Year)	Data Type (Number of Input Variable × Lagged Time)	Output	Method	Performance Measure
Kara et al. (2011) [33]	Turkey ISE National 100 Index (10 × 1)	Direction of stock market (up/down)	ANN, SVM	Directional accuracy
Enke and Mehdiyev (2013) [34]	US S&P 500 index (20 × 1)	Stock price	Fuzzy clustering + fuzzy NN	RMSE
Kristjanpoller et al. (2014) [35]	3 Latin-American stock exchange indices (4 × 2)	Volatility	ANN + GARCH	RMSE, MSE, MAE, MAPE
Yu et al. (2014) [25]	China SSE (7 × 1)	Return rank (divided by 25%)	PCA + SVM	Accuracy (classified by return rank)
Nayak et al. (2015) [36]	India BSE and CNX (11× 1)	Stock index	KNN + SVM	MSE, RMSE, MAPE
Chen and Hao (2017) [26]	China SSE and SZSE (14 × [1~30])	Direction of return (profit/loss)	IG+SVM+KNN	Directional accuracy
Chong et al. (2017) [21]	Korea KOSPI 38 stock returns (38 × 10)	Stock return	DNN	NMSE, RMSE, MAE
Lei (2018) [37]	China SSE Composite Index, CSI 300 Index, Japan Nikkei 225 Index, and US Dow Jones Index (15 × 1)	Stock price	Rough set + Wavelet Neural Network	RMSE, MAD, MAPE, CP, CD

شکل ۲۳

7. پیشنهادات

برای بهبود نتایج و روش می‌توان مدل LSTM-CNN استفاده کرد که در واقع در این مدل الگوهای تصاویر نمودار که با استفاده از روش‌های تکنیکی و فنی بدست می‌آیند را با استفاده از شبکه عصبی پیچشی و ویژگی‌های زمانی موجود در داده‌های سری زمانی مالی برای قیمت‌ها و حجم معاملات بدست آمده و نتیجه نهایی از ترکیب این دو روش یعنی روش تجزیه و تحلیل تکنیکی و فنی و تجزیه و تحلیل سری زمانی حاصل می‌شود.

مورد بعدی که برای بهبود روش، پیشنهاد می‌شود استفاده از الگوریتم GA (Genetic Algorithm) می‌باشد که شامل اپراتورهایی است که از اصول ژنتیکی طبیعی استفاده می‌کنند. در واقع برای پیدا کردن تعداد نرون‌های مناسب در هر لایه به جای آزمون و خطا از این الگوریتم استفاده می‌شود.

به عبارتی در اینجا به جای انجام روش بهینه سازی تصادفی از فرایند تکامل طبیعی الهام می‌گیرد. ویژگی اصلی GA استفاده از جمعیت "کروموزوم" است. هر کروموزوم به عنوان یک راه حل بالقوه برای یک مسئله هدف عمل می‌کند و معمولاً به صورت رشته‌های دو دویی بیان می‌شود. این کروموزوم‌ها بطور تصادفی تولید می‌شوند و آن یکی که راه حل بهتری را ارائه می‌دهد شانس بیشتری برای تولید مثل پیدا می کند.

پیشنهاد بعدی ترکیب مدل LSTM با ARIMA و یا GARCH می‌باشد که در مورد اول علاوه بر مدل‌های غیرخطی، مدل‌های خطی (اگر LSTM جوری تعریف شود که فقط غیر خطی عمل کند) نیز آموزش داده می‌شوند و در مورد دوم نیز ممکن است نتایج کلی تر و بهتری حاصل شود.

همچنین می‌توان با استفاده از مدل Multi variate علاوه بر قیمت بسته شده سهم، قیمت‌های دیگر مانند قیمت باز شده سهم و بالاترین قیمت و پایین ترین قیمت و ... را نیز پیش بینی کرد و زمان مناسب برای فروش سهم در روز را بهتر پیدا کرد.

مدل‌های متناسب دیگری که بعضی از آن‌ها در بخش ۴,۲ بیان شده را می‌توان روی داده‌ها پیاده کرد و شاید به این صورت نتیجه بهتری حاصل شود.

مورد بعدی در نظر گرفتن و تحلیل نظرات و احساسات و متن‌های موجود در رابطه با سهم‌ها می‌باشد.

به طور کلی تحلیل پرتفوی و به کارگیری روش‌های پردازش زبان و غیره در پیش بینی بازده، کشف تقلب و کلاه برداری و تحلیل احساسات و متن‌های حاصل از نظرات سهامداران و ... در تصمیم گیری یاری دهنده می‌باشد، و برای بهبود نتایج حاصل از پیش بینی در نظر گرفتن و انجام آن‌ها پیشنهاد می‌شود.

8. منابع

بیشترین منابعی که در انجام این پروژه مورد استفاده قرار گرفته در زیر بیان شده است.

- https://machinelearningmastery.com/multi-step-time-series-forecasting-long-short-term-memory-networks-python/
- https://analyticsindiamag.com/hands-on-guide-to-lstm-recurrent-neural-network-for-stock-market-prediction/
- https://medium.com/@Ruslan_S_/stock-market-prediction-with-lstm-recurrent-neural-network-4d558f57203f
- https://www.datacamp.com/community/tutorials/lstm-python-stock-market
- https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/
- https://machinelearningmastery.com/make-predictions-long-short-term-memory-models-keras/#:~:text=A%20final%20LSTM%20model%20is,regression%20(a%20real%20value).
- Stock Price Correlation Coefficient Prediction with ARIMA-LSTM Hybrid Model
- PREDICTION AVERAGE STOCK PRICE MARKET USING LSTM
- Using LSTM in Stock prediction and Quantitative Trading
- Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data
- Forecasting stock prices with long-short term memory neural network based on attention mechanism
- LSTM-RNN Automotive Stock Price Prediction
- Stock Market Prediction Using LSTM Recurrent Neural Network
- Stock Price Prediction Using LSTM on Indian Share Market
- Genetic Algorithm-Optimized Long Short-Term Memory Network for Stock Market Prediction
- https://machinelearningmastery.com/convert-time-series-supervised-learning-problem-python/