|  |  |  |
| --- | --- | --- |
|  | **به نام خدا**  **دانشگاه تهران**  **دانشکده‌ مهندسی برق و کامپیوتر** |  |
| **درس شبکه‌های عصبی و یادگیری عمیق**  **تمرین چهارم** | | |

|  |  |
| --- | --- |
| نام و نام خانوادگی | **سارا رستمی – محمدامین شاهچراغی** |
| شماره دانشجویی | **810۱00۳۵۵ - 810۱۹۹۱۹۶** |
| تاریخ ارسال گزارش | **۱۴۰۱.۱۰.۰6** |

**فهرست**

[**پاسخ 1**. **تخمین آلودگی هوا** 5](#_Toc122867837)

[۱-۱. سوالات تشریحی 5](#_Toc122867838)

[۲-۱. دیتاست 6](#_Toc122867839)

[۳-۱. پیش‌پردازش 7](#_Toc122867840)

[۱-۳-۱. Missing Value 7](#_Toc122867841)

[۲-۳-۱. Encoding Categorical Variable 8](#_Toc122867842)

[۳-۳-۱. Normalization 10](#_Toc122867843)

[۴-۳-۱. Pearson Correlation 10](#_Toc122867844)

[۵-۳-۱. Feature Selection 11](#_Toc122867845)

[۶-۳-۱. Supervised dataset 11](#_Toc122867846)

[۴-۱. آموزش شبکه 12](#_Toc122867847)

[**پاسخ ۲** **.** تشخیص اخبار جعلی 9](#_Toc122867848)1

[۱-۲. توضیحات مدل‌ها 9](#_Toc122867849)1

[۲-۲. ورودی مدل 9](#_Toc122867850)1

[۲-۳. پیاده‌سازی مدل](#_Toc122867851) 20

[۱-۳-۲. پیش‌پردازش 20](#_Toc122867852)

[۲-۳-۲. آموزش مدل‌ها](#_Toc122867853) 21

[۴-۲. تحلیل نتایج 24](#_Toc122867854)

**شکل‌ها**

شکل 1- فرمول روش Linear Interpolation 5

شکل 2- روش Linear Interpolation 5

شکل 3 - فرمول pearson correlation 6

شکل 4- فرمول R2 6

شکل 5- داده های ایستگاه Aotizhongxin 7

شکل 6 جایگذاری مقادیر Nan ایستگاه Aotizhongxin 8

شکل 7 – جایگذاری مقادیر Nan ایستگاه در همه ایستگاه ها 8

شکل 8- مقادیر تبدیل جهت به زاویه 22

شکل 9 تبدیل جهت به زاویه 9

شکل 10- داده های نرمالسازی شده 10

شکل 11 – نقشه حرارتی همبستگی داده ها 11

شکل 12- ابعاد داده ها برای lag 1 روز 12

شکل 13- ابعاد داده ها برای lag 7 روز 12

شکل 14- کد ساخت مدل 13

شکل 15 – لایه های مدل برای ورودی با lag 1روز 13

شکل 16 – lossدر هر ایپاک با lag 1 روز 14

شکل 17- آماره ها برای lag 1 روز 14

شکل 18- پیشبینی آلودگی توسط مدل با lag 1 15

شکل 19- summary مدل برای ورودی با lag 7 روز 16

شکل 20- lossدر هر ایپاک با lag 7 روز 17

شکل 21- آماره ها برای lag 7 روز 17

شکل 22 – پیشبینی آلودگی توسط مدل با lag 7 18

شکل 23- خواندن داده‌ها و حذف ستون‌های غیرمرتبط 20

شکل 24- توابع مربوط به تمیزسازی متن 20

شکل 25- مشخصات مدل Hybrid 21

شکل 26- نمودار دقت مدل Hybrid 21

شکل 27- نمودار loss مدل Hybrid 22

شکل 28- خلاصه‌ی عملکرد مدل Hybrid روی داده‌های تست 22

شکل 29- مشخصات مدل RNN 22

شکل 30- نمودار دقت مدل RNN 23

شکل 31- نمودار loss مدل RNN 23

شکل 32- خلاصه‌ی عملکرد مدل روی داده‌های تست 23

**جدول‌ها**

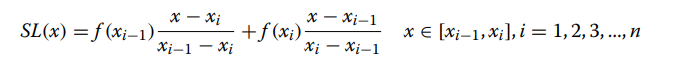
# **پاسخ 1**. **تخمین آلودگی هوا**

۱-۱. سوالات تشریحی

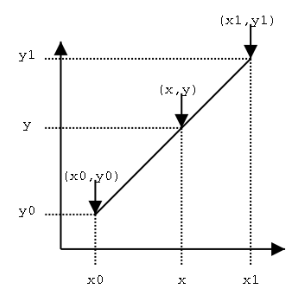
Linear interpolation method :

روش درونیابی خطی یا Linear Interpolation Method روشی است که با استفاده از آن میتوان داده هایی که در یک مجکوعه داده احیانا غایب هستند را تخمین زد معمولا برای تخمین missing values از روش میانگین گیری استفاده میشود اما در داده هایی که فرمت سری زمانی دارند باید از روش هایی مانند Linear Interpolation استفاده برد .

این روش با داشتن 2 نقطه و نرخ تغییرات بین آنها نقطه سومی که بین آنها قرار دارد ولی مقدارش نامشخص است را تخمین میزند به اینصورت که آنها را با یک خط به یکدیگر متصل میکند اگر داده ها خطی نباشند و بین آنها تغییرات بزرگ اتفاق بیافتد تخمین این روش ممکن است دقیق نباشد .



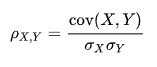
شکل 1- فرمول روش Linear Interpolation



شکل 2- روش Linear Interpolation

Pearson Correlation :

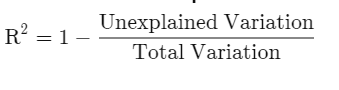
معیاری برای اندازه گیری همبستگی خطی بین 2 مجموعه داده است . در این روش کواریانس دو مجوعه تقسیم بر ضرب انحراف معیار آنها میشود . در واقع کواریانسی است که نرمالسازی شده است . بنابراین همیشه مقدارش بین -1 و 1 است و هرقدر به 1 نزدیکتر باشد یعنی ارتباط و همبستگی قویتری بین دو مجوعه داده وجود دارد .



شکل 3- فرمول pearson correlation

:

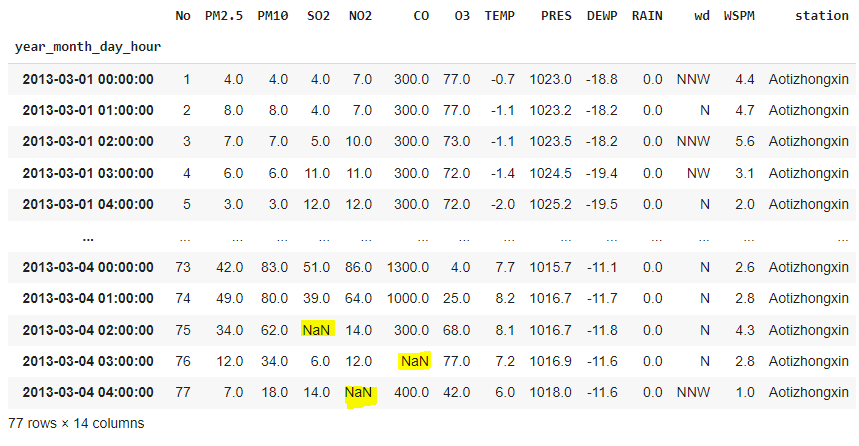
معیار آماری است که نشان میدهد چه نسبتی از واریانس وتغییرات یک متغیر وابسته توسط متغیر یا متغیر های غیروابسته از طریق رابطه رگرسیون توضیح داده میشود . هر چقدر به 1 نزدیکتر یاشد متغیر یا متغیر های غیروابسته ، متغیر وابسته را بهتر توصیف میکنند .



شکل 4- فرمول R2

## ۲-۱. دیتاست

ابتدا همه ی 12 فایل در گوگل درایو بارگذاری میکنیم و سپس از آنجا فرا میخوانیم . در اینجا موقع خواندن داده ها همه ستون های مربوط به نشان دادن زمان و تاریخ را در یک ستون تجمیع میکنیم چون میخواهیم داده ها بر اساس ترتیب زمانی شان مرتب باشند و همه ی داده های مربوط به ساعت و تاریخ همین مساله را نشان میدهند . سپس 77 سطر ابتدایی داده های مربوط به ایستگاه Aotizhongxin را نشان میدهیم . در اینجا چندین خانه داده ها که مقدار ندارند (missing value) مشهود است .

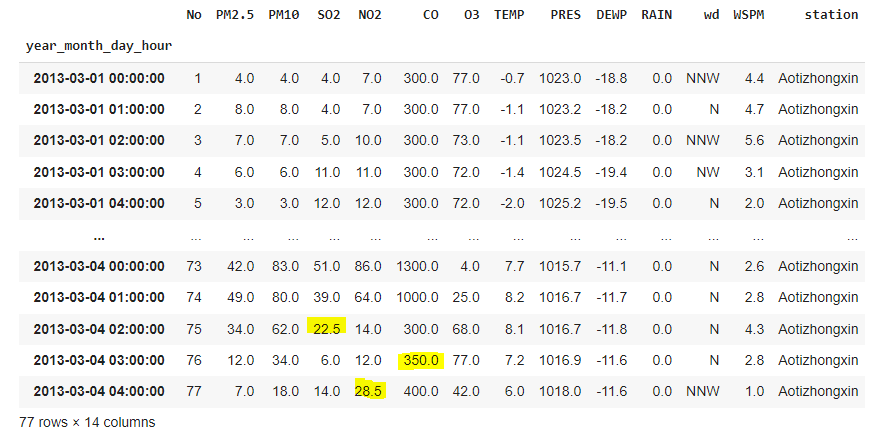


شکل 5- داده های ایستگاه Aotizhongxin

## ۳-۱. پیش‌پردازش

### ۱-۳-۱. Missing Value

داده های غایب را با متد df.interpolate() جایگزین میکنیم . برای دیتا فریم مربوط ایستگاه Aotizhongxin اینکار را میکنیم و 77 سطر اول را نمایش میدهیم تا جایگذاری مقادیر غایب که در شکل 3 با رنگ زرد مشخص شده دیده شود . سپس برای ادامه کارمان کل دیتا فریم که شامل ستون های pm2.5 استگاه های دیگر نیز است را نیز interpolate میکنیم .



شکل 6- جایگذاری مقادیر Nan ایستگاه Aotizhongxin



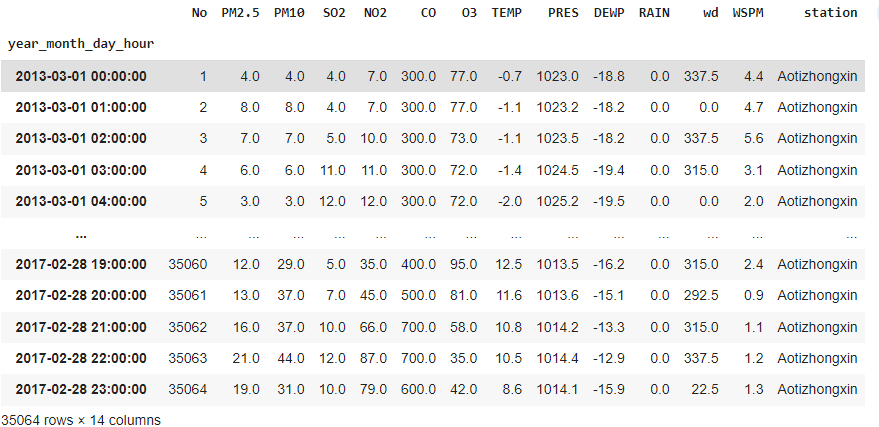
شکل 7- جایگذاری مقادیر Nan ایستگاه در دیتافریم شامل همه ایستگاه ها

### ۲-۳-۱. Encoding Categorical Variable

ستون wd را بر اساس مقاله با اضافه کردن مقادیر 22.5 درجه به عدد تبدیل میکنیم .



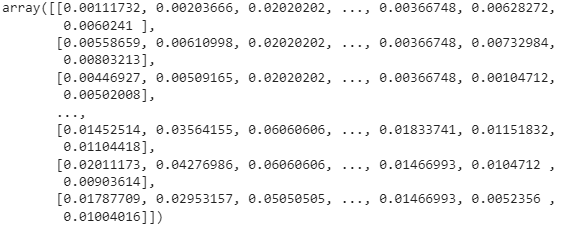
شکل 8- مقادیر تبدیل جهت به زاویه



شکل 9- تبدیل جهت به زاویه

### ۳-۳-۱. Normalization

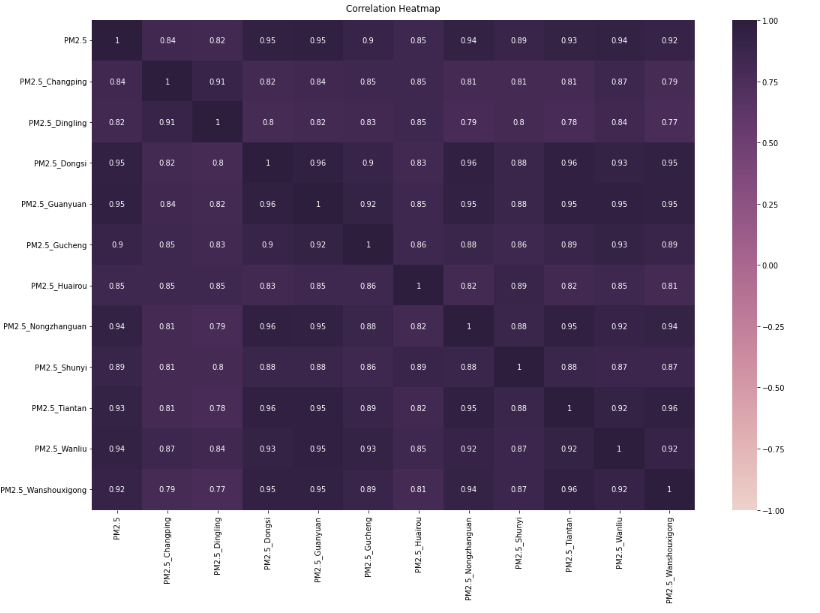
داده ها را با ابتدا به فرمت آرایه در می آوریم به کمک df.values سپس آنها را به کمک MinMaxScaler نرمالسازی میکنیم که نتایج آن را در زیر میبینید .



شکل 10- داده های نرمالسازی شده

### ۴-۳-۱. Pearson Correlation

همبستگی مربوط به Pm2.5 ایستگاه ها را نسبت به هم محاسبه میکنیم و نقشه حرارتی آنرا به نمایش در می آوریم همانطور که دیده میشود مقادیر همبستگی بالایی دارند .



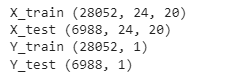
شکل 11- نقشه حرارتی همبستگی داده ها

### ۵-۳-۱. Feature Selection

مقادیر ستون های خواسته شده و ستون pm2.5 مربوط به همه ایستگاه ها را در یک دیتافریم میریزیم که تعداد 20 ویژگی میشود و از آن برای آموزش مدل استفاده میکنیم . این فایل در یک اکسل جداگانه نیز در پوشه تمرین فرستاده شده است .

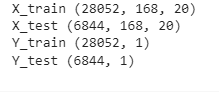
### ۶-۳-۱. Supervised dataset

داده ها را به به حالت supervised در می آوریم یکبار با مقدار 1\*24 = lookback , یکبار با 7\*24 یعنی 168 چون داده ها بر اساس ساعت دسته بندی شده اند این مقادیر lag 1 و 7 روز را برای ما ایجاد میکند . همچنین برچسب ها را که همان ستون ابتدایی است یعنی مقدار pm2.5 جدا میکنیم . سپس داده ها را به نسبت 20 به 80 برای تست و آموزش جدا میکنیم ابعاد داده های تست و آموزش برای lag 1 روز در زیر آمده است .



شکل 12- ابعاد داده ها برای lag 1 روز

اینکار را برای آموزش و تست مدل با lag 7 روز نیز باید انجام دهیم که ابعاد داد ها برای lag 7 روز در زیر آمده است .



شکل 13- ابعاد داده ها برای lag 7 روز

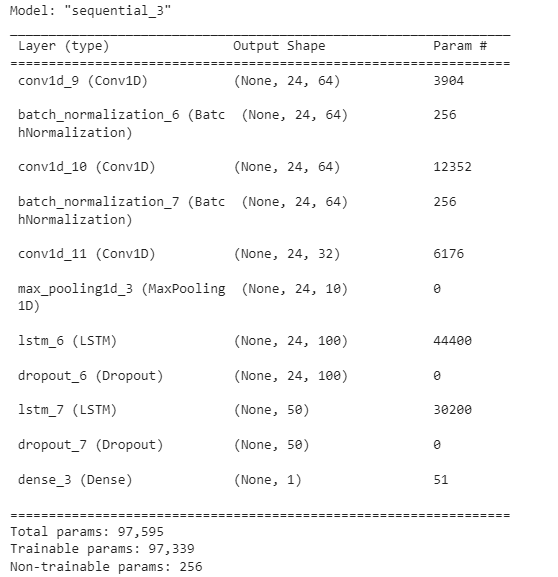
## ۴-۱. آموزش شبکه

مدل را مطابق مقاله به کمک keras میسازیم ابتدا لایه های مربوط به cnn سپس lstm و سپس لایه dense1



شکل 14- کد ساخت مدل

Summary مدل برای lag 1 روز در زیر آمده است .



شکل 15- لایه های مدل برای ورودی با lag 1روز

Learning\_rate و decay باید پایین تنظیم شوند تا مدل به درستی آموزش ببیند .مدل را در 200 ایپاک و batch\_size = 32 آموزش میدهیم و مقدار loss گام به گام مطابق زیر کاهش پیدا میکند .

Epoch 1/200

877/877 [==============================] - 10s 8ms/step - loss: 0.0058

Epoch 2/200

877/877 [==============================] - 7s 8ms/step - loss: 0.0028

Epoch 3/200

877/877 [==============================] - 7s 8ms/step - loss: 0.0020

…

…

Epoch 198/200

877/877 [==============================] - 8s 9ms/step - loss: 2.3763e-04

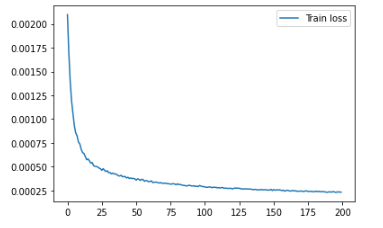
Epoch 199/200

877/877 [==============================] - 7s 8ms/step - loss: 2.3540e-04

Epoch 200/200

877/877 [==============================] - 7s 8ms/step - loss: 2.3553e-04

نمودار loss در هر ایپاک را رسم میکنیم روند کاهشی است .



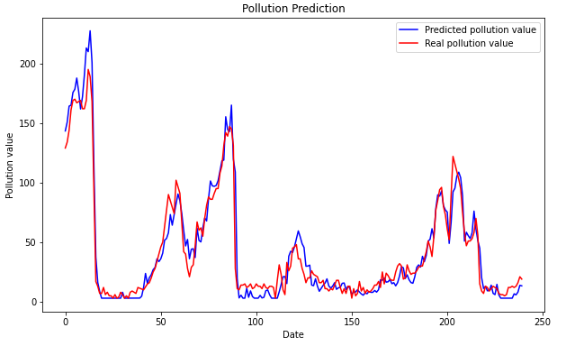
شکل 16- loss در هر ایپاک با lag 1 روز

مدل به شکل مناسبی آموزش میبیند و loss آن کاهش میابد . سپس به کمک مدل آموزش دیده مقادیر را پیشبینی میکنیم و آماره های مورد نظر را برای lag 1روز محاسبه میکنیم که در زیر درج شده است



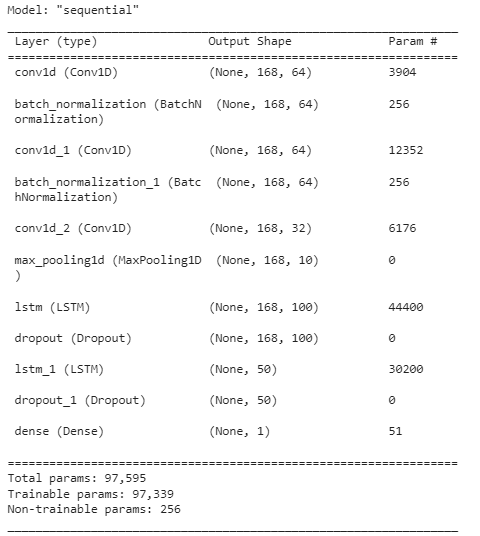
شکل 17- آماره ها برای lag 1 روز

سپس نمودار مربوط به 10 روز از داده های تست که مقدار pm2.5 آنها توسط مدل پیشبینی شده است را رسم میکنیم تا کارایی مدل برای پیشبینی آلودگی را مشاهده کنیم .



شکل 18- پیشبینی آلودگی توسط مدل با lag 1

سپس ورودی ها را برای lag 7 روز supervised و تنظیم میکنیم و مدل را دوباره میسازیم که summary آن در زیر آمده است .



شکل 19- summary مدل برای ورودی با lag 7 روز

مدل را در 200 ایپاک و batch\_size = 32 آموزش میدهیم و مقدار loss گام به گام مطابق زیر کاهش پیدا میکند .

Epoch 1/200

877/877 [==============================] - 10s 11ms/step - loss: 0.0021

Epoch 2/200

877/877 [==============================] - 10s 11ms/step - loss: 0.0017

Epoch 3/200

877/877 [==============================] - 10s 11ms/step - loss: 0.0014

...

...

Epoch 198/200

877/877 [==============================] - 10s 11ms/step - loss: 2.3465e-04

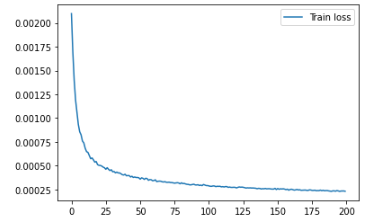
Epoch 199/200

877/877 [==============================] - 10s 11ms/step - loss: 2.3554e-04

Epoch 200/200

877/877 [==============================] - 10s 11ms/step - loss: 2.3242e-04

نمودار loss در هر ایپاک را رسم میکنیم تا روند آنرا ببینیم .



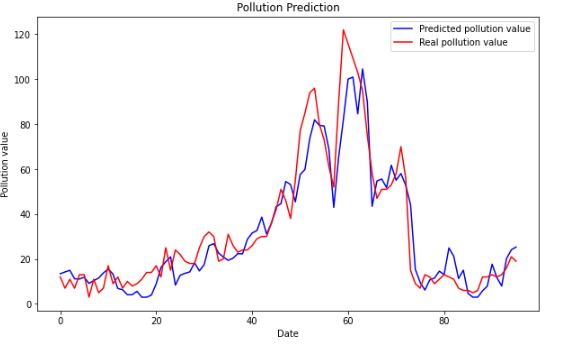
شکل 20- loss در هر ایپاک با lag 7 روز

میبینیم که مدل به خوبی آموزش میبیند . سپس به کمک مدل آموزش دیده مقادیر را پیشبینی میکنیم و آماره های مورد نظر را برای lag 7روز محاسبه میکنیم که در زیر درج شده است .



شکل 21- آماره ها برای lag 7 روز

سپس نمودار مربوط به 10 روز از داده های تست که مقدار pm2.5 آنها توسط مدل با لگ 7 روز پیشبینی شده است را رسم میکنیم تا نحوه ی پیشبینی را روی نمودار ببینیم .



شکل 22- پیشبینی آلودگی توسط مدل با lag 7

lag 1 روز بهتر از lag 7 روز عمل میکند . با بالا رفتن مقدار lag عملکرد مدل تضعیف میشود . و این نشان میدهد که باید از داده هایی که به لحاظ زمانی نزدیکترند برای پیشبینی آلودگی استفاده نمود و داده های قبلی به دلیل تغییر شرایط جوی کمتر برای پیشبینی الودگی مفید اند در کل مدل عملکرد مناسبی برای پیشبینی مقدار pm2.5 دارد و روند کلی میزان این آلاینده را مناسب پیشبینی میکند .

# **پاسخ ۲** **.** تشخیص اخبار جعلی

## ۱-۲. توضیحات مدل‌ها

تفاوت RNN و LSTM :

شبکه‌ های حافظه طولانی کوتاه مدت (Long Short-Term Memory) یک نسخه بهبود یافته از شبکه‌های عصبی بازگشتی(Recurrent Neural Network) هستند که به داده‌ها، «وزن‌هایی» را اختصاص می‌دهند و به RNN این امکان را می‌دهند تا اطلاعات جدید را وارد کند، اطلاعات را فراموش کند و یا به آن ها اهمیت کافی دهد تا روی خروجی اثر بگذارد. در واقع آن ها شبکه‌های RNN را قادر می‌سازند تا ورودی‌ها را در مدت زمان طولانی به خاطر بسپرند. چرا داده های متنی به ویژگی بازگشتی بودن نیاز دارند؟

به این دلیل که در متن ما نیاز به حفظ وابستگی بلند مدت بین کلمات داریم. به این صورت که کلمه اول جمله با کلمه آخر جمله که فعل آن می باشد وابستگی دارد و این وابستگی باید حفظ بشود به همین دلیل از شبکه های بازگشتی استفاده می کنیم که اطلاعات گذشته را حفظ می کنند.

مدل پیشنهادی از CNN برای استخراج ویژگی ‌های محلی(local features) و از LSTM برای یادگیری وابستگی‌های بلندمدت استفاده می‌کند. ابتدا، یک لایه CNN از Conv1D برای پردازش بردارهای ورودی و استخراج ویژگی های محلی که در سطح متن قرار دارند استفاده می شود. خروجی لایه CNN (feature maps ها) ورودی لایه RNN هستند. در واقع لایه RNN از ویژگی های محلی که توسط CNN استخراج شده استفاده می کند و وابستگی های بلندمدت بین کلمات را می آموزد.

## ۲-۲. ورودی مدل

Word embedding ها بردار های عددی هستند که نمایانگر کلمات یک لغت نامه اند. سه تکنیک embedding با نام‌های Word2vec، GloVe و fastText معرفی شده است که در تمام این تکنیک‌ها، شباهت معنایی (semantic similarity) بین کلمات حفظ می‌شود؛ به عبارتی دیگر، با بردارهای به‌دست‌آمده از این تکنیک‌ها، می‌توان معنای کلمات را تشخیص داد و میزان شباهت کلمات مختلف را با یکدیگر به دست آورد. من از روش word2vec استفاده کردم. در این روش به کمک شبکه عصبی یک بردار با اندازه کوچک و ثابت برای نمایش تمام لغات و متون در نظرگرفته شده و با اعداد مناسب در فاز آموزش مدل یا training برای هر لغت این بردار محاسبه می شود برای افزایش دقت این روش، مجموعه داده اولیه که برای آموزش مدل مورد نیاز است، باید حدود چند میلیارد لغت را که درون چندین میلیون سند یا متن به کار رفته اند، در برگیرد. بعد از ایجاد بردارهای مرتبط با هر لغت، برای نمایش برداری هر متن یا خبر ، می توان بردار تک تک کلمات به کار رفته در آنرا یافته و میانگین اعداد هر ستون را به دست آورد که نتیجه آن یک بردار برای هر متن یا سند خواهد بود.

## ۲-۳. پیاده‌سازی مدل

### ۱-۳-۲. پیش‌پردازش

طبق آنچه در مقاله آمده است ابتدا داده را خواندم و جملات متن مقاله هارا را پاکسازی کردم به این معنی که تمام URLها و stop word ها را حذف کردم و سپس داده توکنایز شده را به دو دسته آموزش و تست با نسبت 0.2 تقسیم کردم.



شکل 23- خواندن داده‌ها و حذف ستون‌های غیرمرتبط

توابع مربوط به تمیز‌سازی متن را در شکل زیر مشاهده می‌کنید.



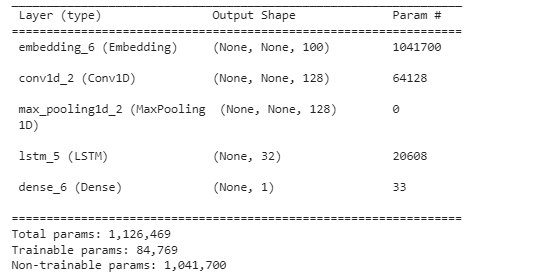
شکل 24- توابع مربوط به تمیزسازی متن

### ۲-۳-۲. آموزش مدل‌ها

داده های پیش پردازش شده را با دو مدل آموزش دادم:

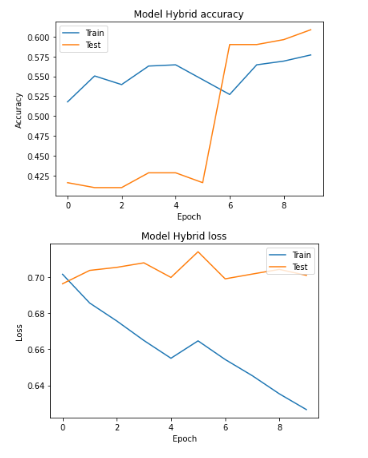
مدل اول : Hybrid(CNN-RNN)

تمام لایه ها و پارامتر ها منطبق بر مقاله پیاده سازی شد.

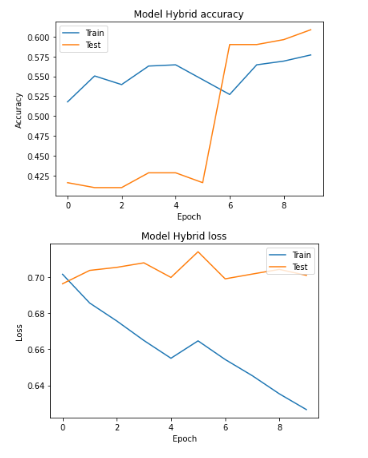


شکل 25- مشخصات مدل Hybrid

مدل نوشته شده را طی 10 ایپاک آموزش دادم که نمودار های دقت و خطا و نتایج مدل در شکل‌های زیر قابل مشاهده می باشد.

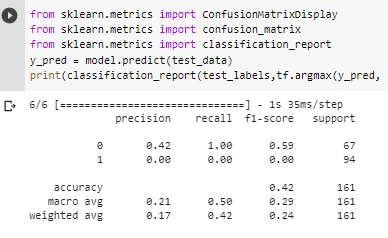


شکل 26- نمودار دقت مدل Hybrid



شکل 27- نمودار loss مدل Hybrid

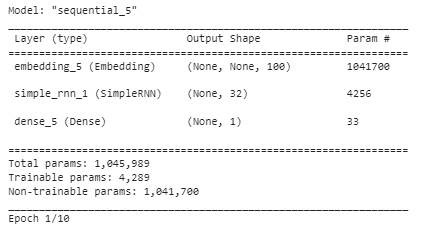
همچنین نتایج F1\_score، recall، Precision و Accuracy این مدل را در شکل زیر مشاهده می‌کنید.



شکل 28 – خلاصه‌ی عملکرد مدل Hybrid روی داده‌های تست

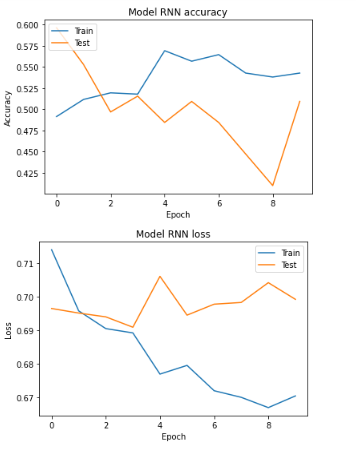
مدل دوم: RNN

مشخصات این مدل را در شکل زیر مشاهده می‌کنید.

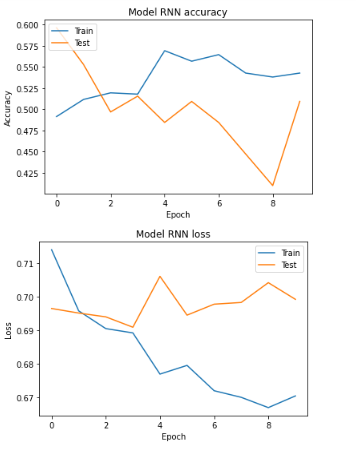


شکل29 - مشخصات مدل RNN

این مدل را هم برای 10 مرحله روی داده های پیش پردازش شده آموزش دادم. نمودار accuracy و loss این مدل را در شکل‌های زیر مشاهده می‌کنید.

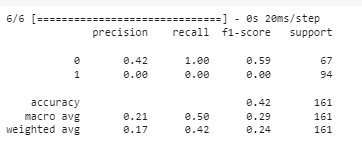


شکل 30- نمودار دقت مدل RNN



شکل 31- نمودار loss مدل RNN

همچنین نتایج F1\_score، recall، Precision و Accuracy این مدل را در شکل زیر مشاهده می‌کنید.



شکل 32- خلاصه‌ی عملکرد مدل روی داده‌های تست

همانطور که مشاهده کردید دقت و خطای مدل Hybrid نسبت به مدل RNN ساده بهبود داشته است و این به دلیل استفاده از شبکه CNN می باشد.

## ۴-۲. تحلیل نتایج

فکر می کنم بزرگترین مشکل این پیاده سازی در کم بودن تعداد داده ها می باشد و اگر از دیتاست بزرگتری استفاده کنیم نتایج بسیار بهتری خواهیم داشت . بهترین نتیجه زمانی پدیدار می شود که دیتاست استفاده شده حجم بالایی داشته باشد و بالانس نیز باشد. تعداد داده های کلاس ها متناسب باشد. اگر تعداد داده های داده 0 بیشتر باشد مدل به سمتی می رود که همه داده هارا 0 پاسخ دهد در صورتی که در این نوع مسائل recall اهمیت بالایی دارد و نمیخواهیم متن فیک تشخیص داده نشده بماند.