



## گزارش پروژه نهایی درس شبکه عصبی

استاد : دکتر عبدوس

سید سروش مجد

شماره دانشجویی

۴۰۰۴۴۳۱۸۱

اسفند ماه ۱۴۰۰

## آماده‌سازی داده‌ها

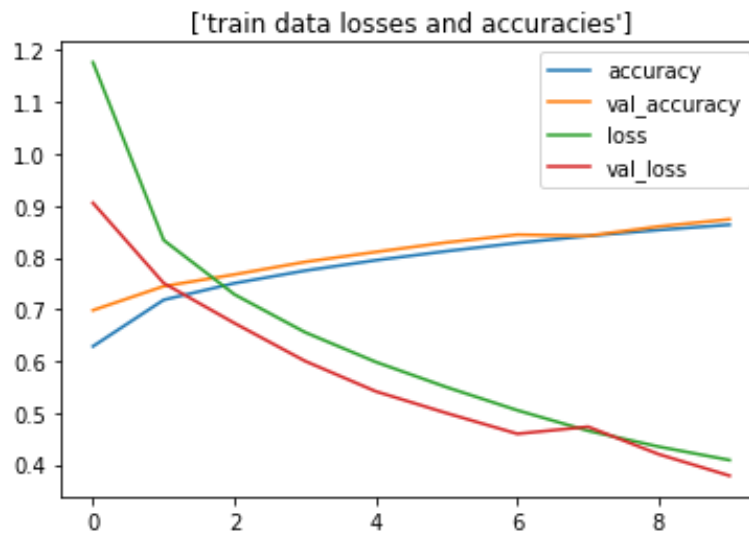
از داده‌های پوشه اول ارائه شده (csh101) برای مسئله استفاده کردیم. به دلیل اینکه لیبل‌ها عددی نبودند ما آن‌ها را عددی کردیم که تعداد این لیبل‌ها ۳۵ عدد بود. جمع کل ستون‌ها ۳۷ تا بود که ستون لیبل‌ها و باقی ستون‌ها را در دو متغیر جداگانه ریختیم. به دلیل اینکه داده‌ها نرمال نبودند و دارای اسکیل متفاوتی بودند (یک سری خیلی بزرگ و یک سری خیلی کوچک بودند) آن‌ها را با تابع MinMaxScaler نرمال‌سازی کردیم. همچنین داده‌هایمان را به صورت sequence درآوردیم، به این صورت که هر ۶۰ ردیف از داده‌ها را به عنوان time step زمانی در نظر گرفتیم. سپس داده‌ها را reshape کرده و به داده‌های آموزش و تست تقسیم‌بندی کردیم.

## مرحله اول

شبکه‌های حافظه‌ی کوتاه‌مدت بلند یا LSTM نوعی از شبکه‌های عصبی بازگشتی هستند که یادآوری داده‌های گذشته در آن‌ها آسان‌تر شده است. مشکل گرادیان کاهشی که در شبکه عصبی بازگشتی وجود داشت نیز در این شبکه‌ها حل شده. شبکه‌های LSTM برای مسائل رده‌بندی، پردازش و پیش‌بینی سری‌های زمانی با استفاده از برجسب‌های زمانی مدت‌های نامعلوم مناسبند. این شبکه مدل را با استفاده از انتشار رو به عقب آموزش می‌دهد. مدل LSTM ما از یک لایه ۱۰۰ یونیتی تشکیل شده است و همچنین دلیل استفاده از dropout این است که تعداد نورون‌های شبکه زیاد است و می‌خواهیم بعضی از نورون‌ها خاموش شود تا دچار over fitting نشویم. همچنین دلیل استفاده از Dense با تعداد ۳۵ unit در لایه آخر این است که ۳۵ لیبل داریم. تابع loss با توجه به multi class بودن مسئله categorical cross entropy است. دقت داده‌های Train و Valid در شکل‌های زیر نشان داده شده است.

```
- loss: 1.1582 - accuracy: 0.6352 - val_loss: 0.8549 - val_accuracy: 0.7123
- loss: 0.8259 - accuracy: 0.7196 - val_loss: 0.7281 - val_accuracy: 0.7480
- loss: 0.7219 - accuracy: 0.7523 - val_loss: 0.6488 - val_accuracy: 0.7733
- loss: 0.6527 - accuracy: 0.7745 - val_loss: 0.6148 - val_accuracy: 0.7863
- loss: 0.6019 - accuracy: 0.7930 - val_loss: 0.5673 - val_accuracy: 0.8037
- loss: 0.5430 - accuracy: 0.8161 - val_loss: 0.5022 - val_accuracy: 0.8311
- loss: 0.4970 - accuracy: 0.8329 - val_loss: 0.4690 - val_accuracy: 0.8379
- loss: 0.4627 - accuracy: 0.8446 - val_loss: 0.4198 - val_accuracy: 0.8574
- loss: 0.4300 - accuracy: 0.8563 - val_loss: 0.3985 - val_accuracy: 0.8676
- loss: 0.4019 - accuracy: 0.8665 - val_loss: 0.3758 - val_accuracy: 0.8740
```

نمودار Accuracy و Loss برای داده‌های Train و Test:



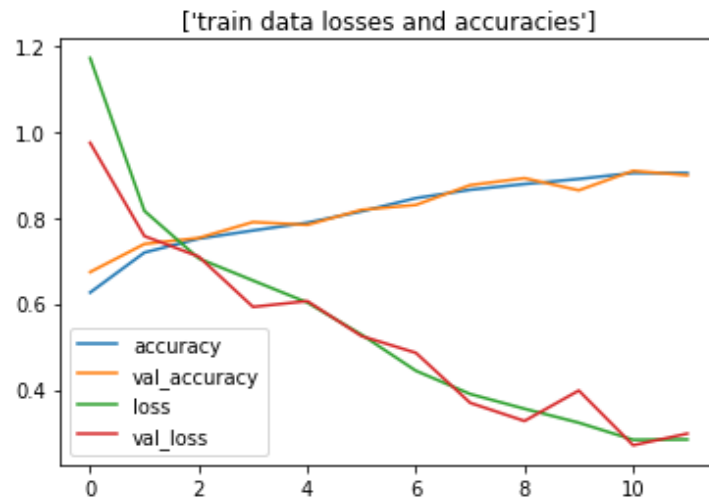
با توجه به اینکه داده‌ها پشت سر هم ضبط شده‌اند بهتر است داده‌ها را به شکل time step به شبکه دادیم.

## مرحله دوم

یکی از کارهایی که شبکه عصبی cnn انجام می‌دهد استخراج ویژگی است. به این معنی که داده‌هایمان را به ابعاد بالاتری منتقل می‌کند و همچنین ویژگی‌های بهتری از داده استخراج می‌کند که با استفاده از آن‌ها می‌توانیم classification بهتری داشته باشیم و انتظار داریم با اعمال یک لایه کانولوشن قبل از لایه LSTM دقت بیشتری نسبت به LSTM خالی داشته باشیم. تعداد feature map های لایه اول ۶۴ است و لایه دوم ۶۴ است. همچنین از یک لایه maxpooling استفاده کردیم تا تعداد ویژگی‌هایمان کاهش پیدا کند. سپس دو لایه LSTM با ۲۰۰ یونیت و یک لایه dense با ۳۵ نورون (۳۵ تا کلاس داریم) قرار دارد. تابع loss با توجه به multy class بودن مسئله categorical cross entropy است.

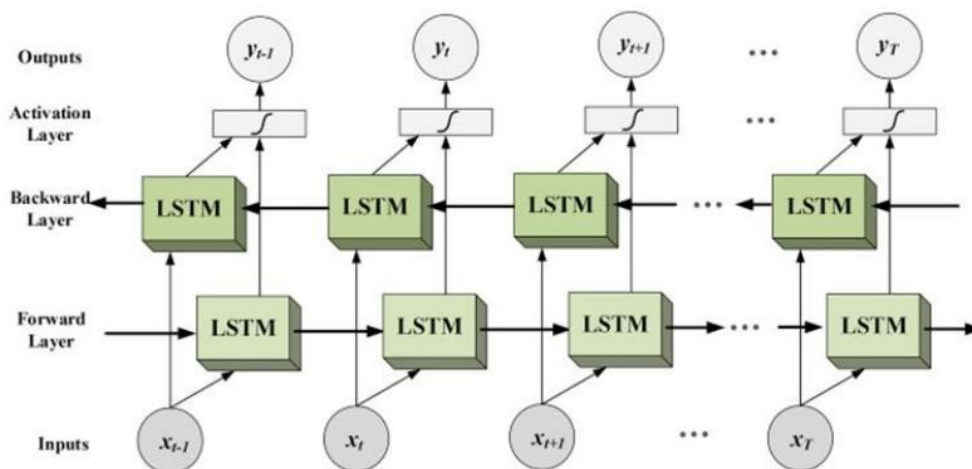
Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 60, 64)	11584
conv1d_1 (Conv1D)	(None, 60, 64)	20544
lstm (LSTM)	(None, 60, 200)	212000
lstm_1 (LSTM)	(None, 200)	320800
dense (Dense)	(None, 35)	7035
=====		
Total params: 571,963		
Trainable params: 571,963		
Non-trainable params: 0		

نمودار Accuracy و Loss برای داده‌های Train و Test:



## مرحله سوم

شبکه LSTM دو طرفه یا Bi-Directional LSTM دقت بهتری نسبت به LSTM ساده مهیا می‌کند و همانند LSTM ساده برای مسائل sequential کاربرد دارد. در این شبکه داده train ورودی از هر دو طرف یعنی "گذشته به آینده" و "آینده به گذشته" وارد می‌شود. در این بخش می‌خواهیم عملکرد این شبکه را بر روی داده‌هایمان بسنجیم و انتظار داریم در نتایج بهبود حاصل شود.



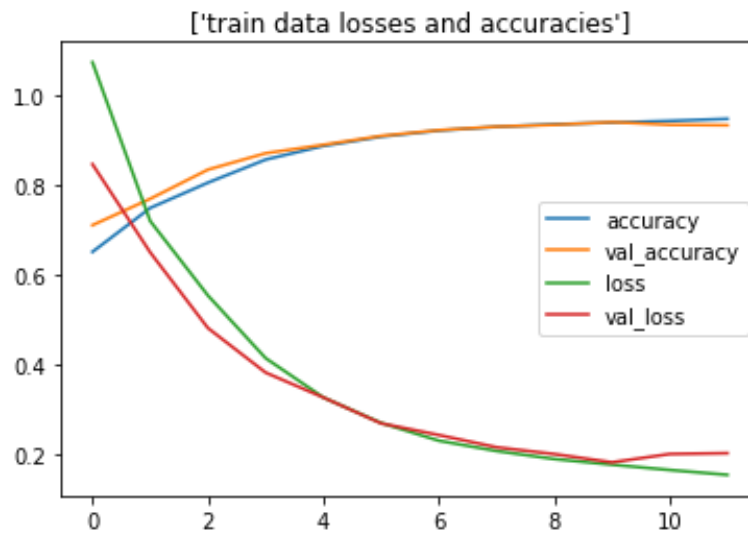
شبکه Bi-Directional LSTM

مدل ما: دولایه کانولوشن ۶۴ تایی، یک لایه max\_pooling، دولایه BI-LSTM و لایه خروجی

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 60, 64)	11584
conv1d_1 (Conv1D)	(None, 60, 64)	20544
max_pooling1d (MaxPooling1D)	(None, 8, 64)	0
bidirectional (Bidirectional)	(None, 8, 512)	657408
bidirectional_1 (Bidirectional)	(None, 512)	1574912
dense (Dense)	(None, 35)	17955
=====		
Total params: 2,282,403		
Trainable params: 2,282,403		
Non-trainable params: 0		

```
- loss: 1.0726 - accuracy: 0.6499 - val_loss: 0.8454 - val_accuracy: 0.7094
- loss: 0.7187 - accuracy: 0.7478 - val_loss: 0.6488 - val_accuracy: 0.7682
- loss: 0.5527 - accuracy: 0.8042 - val_loss: 0.4801 - val_accuracy: 0.8333
- loss: 0.4129 - accuracy: 0.8557 - val_loss: 0.3808 - val_accuracy: 0.8698
- loss: 0.3253 - accuracy: 0.8861 - val_loss: 0.3253 - val_accuracy: 0.8887
- loss: 0.2685 - accuracy: 0.9068 - val_loss: 0.2675 - val_accuracy: 0.9085
- loss: 0.2289 - accuracy: 0.9204 - val_loss: 0.2416 - val_accuracy: 0.9213
- loss: 0.2059 - accuracy: 0.9288 - val_loss: 0.2142 - val_accuracy: 0.9292
- loss: 0.1880 - accuracy: 0.9344 - val_loss: 0.1991 - val_accuracy: 0.9332
- loss: 0.1754 - accuracy: 0.9383 - val_loss: 0.1806 - val_accuracy: 0.9393
- loss: 0.1636 - accuracy: 0.9419 - val_loss: 0.1992 - val_accuracy: 0.9334
```

نمودار Accuracy و Loss برای داده‌های Train و Test:



مشاهده می‌کنیم که دقت به ۹۳ درصد رسید و این پیشرفت با توجه به داده ۳۰۰ هزار تایی قابل توجه به نظر می‌رسد. دقت بر روی داده‌های تست:

```
1883/1883 [=====] - 50s 22ms/step - loss: 0.1983 - accuracy: 0.9351
test_loss = 0.19832094013690948
test_accuracy = 0.9351235628128052
```

لینک کدها برای مراحل به ترتیب:

[https://colab.research.google.com/drive/19slURCIE1S675p6EMLu\\_baYBgIR5gFaI?usp=sharing&authuser=2#scrollTo=TMtDDp6sLTpb](https://colab.research.google.com/drive/19slURCIE1S675p6EMLu_baYBgIR5gFaI?usp=sharing&authuser=2#scrollTo=TMtDDp6sLTpb)

<https://colab.research.google.com/drive/19JTF8L2Ff6DpQepBrRJ1T1mlzfzcq2UD#scrollTo=Uw1Qzrs7t65>

<https://colab.research.google.com/drive/1F2F4OIeEkrjA5qPhBnc11fAivaTQB-Ax#scrollTo=MCE9PNYN0X7W>