

Final Project – Predict Stock Performance using NLP

Sarah Adilijiang

1. First, load the data and construct the Databunch object.
2. Then, finetune the Language Model, which has following steps:
 - 1) Construct a Language Model Learner using the AWD_LSTM model.
 - 2) Finetune the Language Model Learner to the texts in the 10-K forms in our data by using appropriate learning rate. The fitting result is shown as below.

We can see that the 1st, 4th, and 5th epochs all have the same highest accuracy, which is 0.585714.

```
learn_lm.fit_one_cycle(5, lr, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	1.840808	2.275537	0.585714	03:16
1	1.656230	2.200438	0.571429	03:16
2	1.563748	2.167121	0.600000	03:18
3	1.413669	2.121336	0.585714	03:17
4	1.333790	2.131405	0.585714	03:18

- 3) Save the finetuned Language Model, which will be used as the Encoder in the following classification models.
3. Finally, build classification models for three different time (1 day, 1 week, and 2 weeks) after releasing the 10-K forms. The procedure is similar for three models, which has following steps:
 - 1) Construct a Text Classifier Learner using the AWD_LSTM model.
 - 2) Load the pretrained Language Model as the Encoder for the Text Classifier Learner.
 - 3) Fit the Text Classifier Learner to our 10-K form text data by using appropriate learning rate. The fitting results for three models are shown as below.

a. 1 Day

We can see that the 7th epoch has the highest accuracy, which is 0.489491.

```
learn_c1.fit_one_cycle(10, lr, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	1.162032	1.086996	0.428860	01:10
1	1.087392	1.057042	0.451496	01:10
2	1.029701	1.044737	0.461601	01:09
3	1.007576	1.050638	0.467664	01:08
4	0.988402	1.016719	0.474939	01:09
5	0.974206	1.025467	0.470897	01:09
6	0.950541	1.022957	0.489491	01:08
7	0.932153	1.026013	0.485044	01:09
8	0.923151	1.031669	0.489087	01:07
9	0.902353	1.023295	0.479386	01:08

b. 1 Week

We can see that the 5th epoch has the highest accuracy, which is 0.444220.

```
learn_c2.fit_one_cycle(10, lr, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	1.130805	1.091971	0.420776	01:10
1	1.053501	1.046720	0.437753	01:09
2	1.042528	1.042613	0.436136	01:09
3	1.031996	1.049541	0.431285	01:08
4	1.018730	1.038823	0.444220	01:09
5	1.012812	1.040935	0.434115	01:09
6	0.988639	1.053175	0.427648	01:08
7	0.961860	1.059370	0.427243	01:08
8	0.936594	1.085965	0.430881	01:07
9	0.925221	1.066486	0.426839	01:08

c. 2 Weeks

We can see that the 3th epoch has the highest accuracy, which is 0.470897.

```
learn_c3.fit_one_cycle(10, lr, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	1.088822	1.040307	0.433306	01:10
1	1.017981	1.018048	0.464834	01:09
2	0.993062	0.997974	0.470897	01:09
3	0.983364	1.019694	0.466855	01:08
4	0.972454	1.003678	0.468876	01:09
5	0.971872	1.000222	0.466451	01:09
6	0.946927	1.011721	0.461601	01:08
7	0.916791	1.024168	0.467259	01:08
8	0.889226	1.039986	0.469281	01:07
9	0.878427	1.031930	0.469281	01:08

4. In the end, let's compare the above classification results and make the conclusions.

The accuracies for three classification models are:

- 1) **1 day**: the highest accuracy is 0.489491.
- 2) **1 week**: the highest accuracy is 0.444220.
- 3) **2 weeks**: the highest accuracy is 0.470897.

We can see that the 1-day model has the best performance, i.e. the highest accuracy, followed by the 2-week model. And the 1-week model has the worst performance. However, all the three models have accuracies that are much higher than a random classification model whose accuracy would be only about 0.33.

As a result, we may conclude that the texts in the 10-K forms can be used to predict the performance of the company's stock performance after the release of the 10-K forms. And it provides the best prediction for the stock performance 1 day after releasing 10-K forms.