

Dataset 的部分，我把 src 和 tgt 結合起來(combined)，把 combined 當作 input 送給 model 去 train。target 就是 tgt

	src	tgt	combined
0	0+0=	0	0+0=0
1	0-0=	0	0-0=0
2	0*0=	0	0*0=0
3	(0+0)*0=	0	(0+0)*0=0
4	0+0*0=	0	0+0*0=0

我把全部的 Dataset 中前面 10000 和後面 10000 筆資料中有括號的和沒有括號的另外創 dataset(brackets_df, no_brackets_df)，我覺得有無括號對答案的預測應該蠻重要的，因為有要先算的部分，和一般從左到右算還是有不一樣的，而且有些是有括號和沒括號結果一樣。還有答案是正的和負的個別也創了 dataset，這部分我只是覺得可能答案的正負會影響 model 做正負的運算。而 math_dataset 是我隨便取 20000 筆的資料(263000:283000)，還有一個 total dataset，我取前 10000 和後 10000，這兩個單純是隨便一筆資料來訓練。

brackets_df	no_brackets_df																																																
<table><tr><th></th><th>src</th><th>tgt</th><th>combined</th></tr><tr><td>0</td><td>(0+0)*0=</td><td>0</td><td>(0+0)*0=0</td></tr><tr><td>1</td><td>(0-0)*0=</td><td>0</td><td>(0-0)*0=0</td></tr><tr><td>2</td><td>0+(0*0)=</td><td>0</td><td>0+(0*0)=0</td></tr><tr><td>3</td><td>0-(0*0)=</td><td>0</td><td>0-(0*0)=0</td></tr><tr><td>4</td><td>0*(0+0)=</td><td>0</td><td>0*(0+0)=0</td></tr></table>		src	tgt	combined	0	(0+0)*0=	0	(0+0)*0=0	1	(0-0)*0=	0	(0-0)*0=0	2	0+(0*0)=	0	0+(0*0)=0	3	0-(0*0)=	0	0-(0*0)=0	4	0*(0+0)=	0	0*(0+0)=0	<table><tr><th></th><th>src</th><th>tgt</th><th>combined</th></tr><tr><td>0</td><td>0+0=</td><td>0</td><td>0+0=0</td></tr><tr><td>1</td><td>0-0=</td><td>0</td><td>0-0=0</td></tr><tr><td>2</td><td>0*0=</td><td>0</td><td>0*0=0</td></tr><tr><td>3</td><td>0+0*0=</td><td>0</td><td>0+0*0=0</td></tr><tr><td>4</td><td>0-0*0=</td><td>0</td><td>0-0*0=0</td></tr></table>		src	tgt	combined	0	0+0=	0	0+0=0	1	0-0=	0	0-0=0	2	0*0=	0	0*0=0	3	0+0*0=	0	0+0*0=0	4	0-0*0=	0	0-0*0=0
	src	tgt	combined																																														
0	(0+0)*0=	0	(0+0)*0=0																																														
1	(0-0)*0=	0	(0-0)*0=0																																														
2	0+(0*0)=	0	0+(0*0)=0																																														
3	0-(0*0)=	0	0-(0*0)=0																																														
4	0*(0+0)=	0	0*(0+0)=0																																														
	src	tgt	combined																																														
0	0+0=	0	0+0=0																																														
1	0-0=	0	0-0=0																																														
2	0*0=	0	0*0=0																																														
3	0+0*0=	0	0+0*0=0																																														
4	0-0*0=	0	0-0*0=0																																														
positive_df	negative_df																																																
<table><tr><th></th><th>src</th><th>tgt</th><th>combined</th></tr><tr><td>0</td><td>0+0=</td><td>0</td><td>0+0=0</td></tr><tr><td>1</td><td>0-0=</td><td>0</td><td>0-0=0</td></tr><tr><td>2</td><td>0*0=</td><td>0</td><td>0*0=0</td></tr><tr><td>3</td><td>(0+0)*0=</td><td>0</td><td>(0+0)*0=0</td></tr><tr><td>4</td><td>0+0*0=</td><td>0</td><td>0+0*0=0</td></tr></table>		src	tgt	combined	0	0+0=	0	0+0=0	1	0-0=	0	0-0=0	2	0*0=	0	0*0=0	3	(0+0)*0=	0	(0+0)*0=0	4	0+0*0=	0	0+0*0=0	<table><tr><th></th><th>src</th><th>tgt</th><th>combined</th></tr><tr><td>0</td><td>0-0-1=</td><td>-1</td><td>0-0-1=-1</td></tr><tr><td>1</td><td>0+0-1=</td><td>-1</td><td>0+0-1=-1</td></tr><tr><td>2</td><td>(0+0)-1=</td><td>-1</td><td>(0+0)-1=-1</td></tr><tr><td>3</td><td>0+(0-1)=</td><td>-1</td><td>0+(0-1)=-1</td></tr><tr><td>4</td><td>0-(0+1)=</td><td>-1</td><td>0-(0+1)=-1</td></tr></table>		src	tgt	combined	0	0-0-1=	-1	0-0-1=-1	1	0+0-1=	-1	0+0-1=-1	2	(0+0)-1=	-1	(0+0)-1=-1	3	0+(0-1)=	-1	0+(0-1)=-1	4	0-(0+1)=	-1	0-(0+1)=-1
	src	tgt	combined																																														
0	0+0=	0	0+0=0																																														
1	0-0=	0	0-0=0																																														
2	0*0=	0	0*0=0																																														
3	(0+0)*0=	0	(0+0)*0=0																																														
4	0+0*0=	0	0+0*0=0																																														
	src	tgt	combined																																														
0	0-0-1=	-1	0-0-1=-1																																														
1	0+0-1=	-1	0+0-1=-1																																														
2	(0+0)-1=	-1	(0+0)-1=-1																																														
3	0+(0-1)=	-1	0+(0-1)=-1																																														
4	0-(0+1)=	-1	0-(0+1)=-1																																														

Model 部分，因為我是把整個算式丟給 model 去算，所以預測出來的 pred_y 也是整個算式字元的預測，但我只要等號後面預測的答案，所以我從 batch_x 中找出等號所在的 index，把後面的拿去算 loss。loss function 沒有改變，還是用 Cross Entropy。

total	brackets
<pre> 4 * 1 5 = pred_y:- 1 6 batch_y:6 0 4 9 * 4 + 1 4 = pred_y:2 9 6 batch_y:2 1 0 4 - 1 5 + 0 = pred_y:1 9 6 9 batch_y:1 9 (1 5 * 0) - 4 = pred_y:3 0 batch_y:- 4 4 - 1 5 - 0 = pred_y:9 0 5 5 2 batch_y:1 9 Validation Loss: 8.437071301159449e-06 </pre>	<pre> (4 + 1 5) * 1 = pred_y:5 5 batch_y:1 9 4 - 1 5 * 0 = pred_y:2 batch_y:4 Validation Loss: 3.300430762465112e-06 </pre>
no_brackets	positive
<pre> (4 - 1 4) + 4 9 = pred_y:4 2 batch_y:3 9 4 + (1 4 - 4 9) = pred_y:8 8 8 batch_y:- 3 1 4 - 1 4 + 4 8 = pred_y:1 1 9 batch_y:3 8 4 - 1 4 + 4 9 = pred_y:7 8 1 batch_y:3 9 Validation Loss: 1.938885134222801e-06 </pre>	<pre> 4 + 1 5 + 0 = pred_y:1 7 3 0 batch_y:1 9 0 * (4 - 1 5) = pred_y:2 batch_y:0 (4 + 1 4) - 4 9 = pred_y:4 2 1 4 batch_y:- 3 1 (4 + 1 5) - 0 = pred_y:5 0 batch v:1 9 Validation Loss: 1.2610510111699114e-06 </pre>
negative	math
<pre> 4 9 * (4 + 1 4) = pred_y:- 1 8 batch_y:8 8 2 4 + 1 4 + 4 9 = pred_y:- 1 3 batch_y:6 7 1 * (4 - 1 5) = pred_y:- 4 0 batch_y:- 1 1 4 - 1 5 - 0 = pred_y:- 1 0 batch_y:- 1 1 Validation Loss: 7.280849558810587e-07 </pre>	<pre> 4 - 1 4 - 4 8 = pred_y:4 2 9 batch_y:- 5 8 4 - 1 5 = pred_y:3 6 batch_y:- 1 1 4 + (1 4 - 4 9) = pred_y:2 3 3 batch_y:- 3 1 4 + 1 4 - 4 9 = pred_y:1 7 9 batch_y:- 3 1 Validation Loss: 3.3517582664899237e-07 </pre>

嘗試調整

```
4 + 1 4 - 4 9 =  
pred_y:- 3 2  
batch_y:- 3 1  
( 4 + 1 5 ) * 1 =  
pred_y:1 7  
batch_y:1 9  
( 1 4 * 4 8 ) - 4 =  
pred_y:3 0  
batch_y:6 6 8  
4 * 1 4 * 4 9 =  
pred_y:0  
batch_y:2 7 4 4  
1 * ( 4 - 1 5 ) =  
pred_y:0  
batch_y:- 1 1
```

batch_size = 1000

Validation Loss: 0.03807859867811203

batch size 比較大，loss 也變大了

```
4 + 1 4 * 4 9 =  
pred_y:1 5 5  
batch_y:6 9 0  
( 4 + 1 4 ) - 4 9 =  
pred_y:- 2 0  
batch_y:- 3 1  
4 9 * ( 4 + 1 4 ) =  
pred_y:1 8 2  
batch_y:8 8 2  
( 1 4 * 4 8 ) - 4 =  
pred_y:4 3  
batch_y:6 6 8  
4 + 1 4 + 4 8 =  
pred_y:5 4  
batch_y:6 6
```

在一個 epoch 中依序用不同資料訓練

Validation Loss: 2.143628563544553e-07

雖然 loss 看起來變更小了，但錯誤還是蠻多的