1. Model analysis

   In this Project 2.a assignment, the model I used consists of an embedding layer that converts each tokenized word into input vectors, which are then fed into an RNN layer. For the subsequent RNN layer, a single layer is sufficient for training, as the task involves simple arithmetic operations with three numbers (addition, subtraction, multiplication, and division). Finally, a fully connected layer compresses the output into a single class. For the loss function, I chose mean squared error to address the regression problem in this assignment. The optimizer used is Adam.

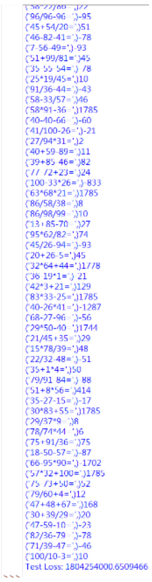
2. Dataset analysis

   I designed two datasets, each containing 100,000 samples. The first dataset consists of 90% addition and subtraction operations with three numbers, and 10% includes addition, subtraction, multiplication, and division operations. The second dataset contains only addition and subtraction operations, with 50% consisting of three two-digit numbers and 50% of three one-digit numbers.

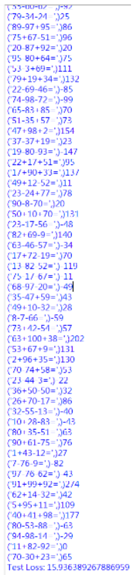
   I believe that the quantity of data significantly impacts training results. For instance, the predictions for the addition and subtraction operations in the first dataset are likely to be better than those for the mixed operations. This is because the model is less familiar with the multiplication and division operations, leading to inherently poorer predictions compared to the more common addition and subtraction operations.

Left: Test results for 10,000 samples of multiplication and division with three numbers.

Right: Test results for 10,000 samples of addition and subtraction with three numbers.



```
('96/96-96=',)-95
('45+54/20=',)51
('46-82-41=',)-78
('7-56-49=',)-93
('51+99/81=',)45
('35-55-54=',)-78
('25*19/45=',)10
('91/36-44=',)-43
('58-33/57=',)46
('58*91-36=',)1785
('40-40-66=',)-60
('41/100-26=',)-21
('27/94*31=',)2
('40+59-89=',)11
('39+85-46=',)82
('77-72+23=',)24
('100-33*26=',)-833
('63*68*21=',)1785
('86/58/38=',)8
('86/98/99=',)10
('13+85-70=',)27
('95*62/82=',)74
('45/26-94=',)-93
('20+26-5=',)45
('32*64+44=',)1778
('36-19*1=',)-21
('42*3+21=',)129
('83*33-25=',)1785
('40-26*41=',)-1287
('68-27-96=',)-56
('29*50-40=',)1744
('21/45+35=',)29
('15*78/39=',)48
('22/32-48=',)-51
('35+1*4=',)50
('79/91-84=',)-88
('51+8*56=',)414
('35-27-15=',)-17
('30*83+55=',)1785
('29/37*9=',)8
('78/74*44=',)6
('75+91/36=',)75
('18-50-57=',)-87
('66-95*90=',)-1702
('57*32+100=',)1785
('75-73+50=',)52
('79/60+4=',)12
('47+48+67=',)168
('30+39/29=',)20
('47-59-10=',)-23
('82/36-79=',)-78
('71/39-47=',)-46
('100/10-3=',)10
Test Loss: 1804254000.6509466
```



```
('79-34-24=',)25
('89-97+95=',)86
('75+67-51=',)96
('20-87+92=',)20
('95-80+64=',)75
('53-3+69=',)111
('79+19+34=',)132
('22-69-46=',)-85
('74-98-72=',)-99
('65-83+85=',)70
('51-35+57=',)73
('47+98+2=',)154
('37-37+19=',)23
('19-80-93=',)-147
('22+17+51=',)95
('17+90+33=',)137
('49+12-52=',)11
('23-24+77=',)78
('90-8-70=',)20
('50+10+70=',)131
('23-17-56=',)-48
('82+69-9=',)140
('63-46-57=',)-34
('17+72-19=',)70
('13-82-52=',)-119
('75-17-67=',)-11
('68-97-20=',)-49
('35-47+59=',)43
('49+10-32=',)28
('8-7-66=',)-59
('73+42-54=',)57
('63+100+38=',)202
('53+67+9=',)131
('2+96+35=',)130
('70-74+58=',)53
('23-44-3=',)-22
('36+50-50=',)32
('26+70-17=',)86
('32-55-13=',)-40
('10+28-83=',)-43
('80+35-51=',)63
('90+61-75=',)76
('1+43-12=',)27
('7-76-9=',)-82
('97-76-62=',)-43
('91+99+92=',)274
('62+14-32=',)42
('5+95+11=',)109
('40+41+98=',)177
('80-53-88=',)-63
('94-98-14=',)-29
('11+82-92=',)0
('70-30+23=',)65
Test Loss: 15.936389267886959
```

Secondly, the range of the data results can impact training outcomes. For instance, the predictions for the one-digit operations in the second dataset are likely to be better than those for the two-digit operations. This is because datasets with a larger range of results require a greater volume of data to train the model on a more diverse set of combinations. As a result, the predictions for two-digit operations tend to be poorer than those for one-digit operations.

Left: Test results for 10,000 samples of three two-digit numbers.

Right: Test results for 10,000 samples of three one-digit numbers.

```
('93-68-34=',)-9
('82-21+81=',)142
('74-16-41=',)18
('71-22-39=',)10
('44-63-39=',)-58
('37+41-13=',)65
('60-84+78=',)54
('82+58+27=',)167
('79+46-59=',)67
('54-18+84=',)121
('12+65-53=',)24
('76+84+17=',)177
('48-27+98=',)119
('54-32+84=',)106
('95+33-52=',)76
('24-41+44=',)27
('26-10-21=',)-5
('75-93-70=',)-87
('33-43-57=',)-67
('42+45+17=',)104
('41-18+28=',)52
('96+27+13=',)136
('39+48+40=',)127
('39+12-28=',)22
('71-81-13=',)-22
('38+10+87=',)135
('48-66-51=',)-69
('54+87-48=',)94
('49-59-21=',)-31
('26-52+78=',)53
('33+42-50=',)24
('85-49-25=',)12
('31+91+67=',)189
('48+73-59=',)62
('75-67-86=',)-78
('10+14-77=',)-52
('27-69+40=',)-2
('17+60+96=',)172
('78-48+68=',)99
('36-42-23=',)-29
('14-49+63=',)27
('82-23+98=',)157
('83-47-42=',)-6
('53-46-24=',)-18
('77+46-69=',)54
('17-52+46=',)11
('24-53+92=',)63
('96+46-27=',)115
('90-47+23=',)66
('27+49+21=',)98
('69+95-42=',)121
('68-71-84=',)-87
('41-82+46=',)5
Test Loss: 0.3379486189672927
```

```
('4-3-4=',)-3
('8+9+8=',)24
('2+5-2=',)5
('2+9+0=',)10
('1-0+0=',)1
('6-9-2=',)-5
('3-4+4=',)3
('0-7-4=',)-11
('2-0+8=',)10
('5-0-8=',)-3
('7-0+8=',)15
('3+4-2=',)5
('8+7-8=',)7
('1+6-7=',)0
('5-0+2=',)7
('7+2-5=',)4
('4-9+7=',)2
('7-1+9=',)15
('3-4-1=',)-2
('2+7+2=',)11
('9+1-2=',)8
('6-5-6=',)-5
('1+8-9=',)0
('3-3+5=',)5
('6+8+8=',)21
('9+9-5=',)13
('5-3+7=',)9
('8+2-4=',)6
('5-2+3=',)6
('6-7+5=',)4
('7-0-1=',)6
('7+3-0=',)10
('7-9-9=',)-11
('0+9-7=',)2
('8+6-8=',)6
('4+4+4=',)12
('5+9-9=',)5
('6+3+0=',)9
('0+0-4=',)-4
('7-8-2=',)-3
('0-8-9=',)-17
('2+9+1=',)12
('3+5+5=',)13
('1+4-9=',)-4
('8+7-7=',)8
('0-1-4=',)-5
('5+2-1=',)6
('0-1+4=',)3
('2+7+4=',)13
('8+3+2=',)13
('7+7-5=',)9
('2+0-6=',)-4
('8+8-0=',)16
Test Loss: 0.06415668916736326
```

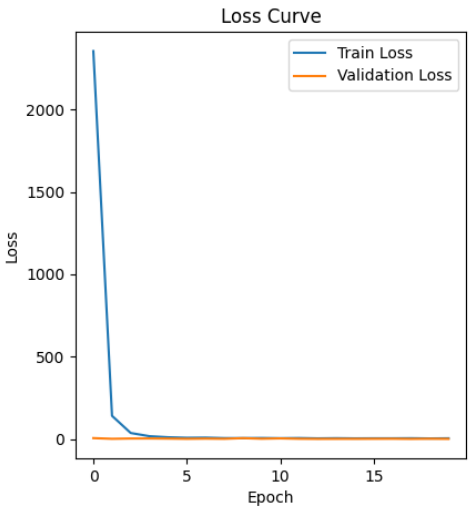
3. Discussion

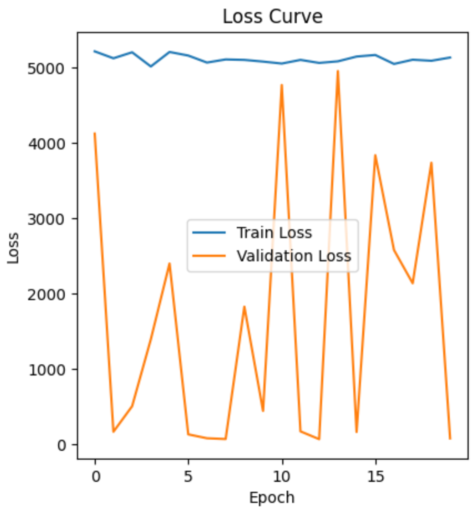
In model training, a larger learning rate makes it more difficult for the gradient to converge to the minimum value and can lead to oscillations. However, the training process will be faster. Conversely, a smaller learning rate facilitates convergence of the gradient to the minimum value, but results in a slower training process.

Left image: Learning rate = 0.001

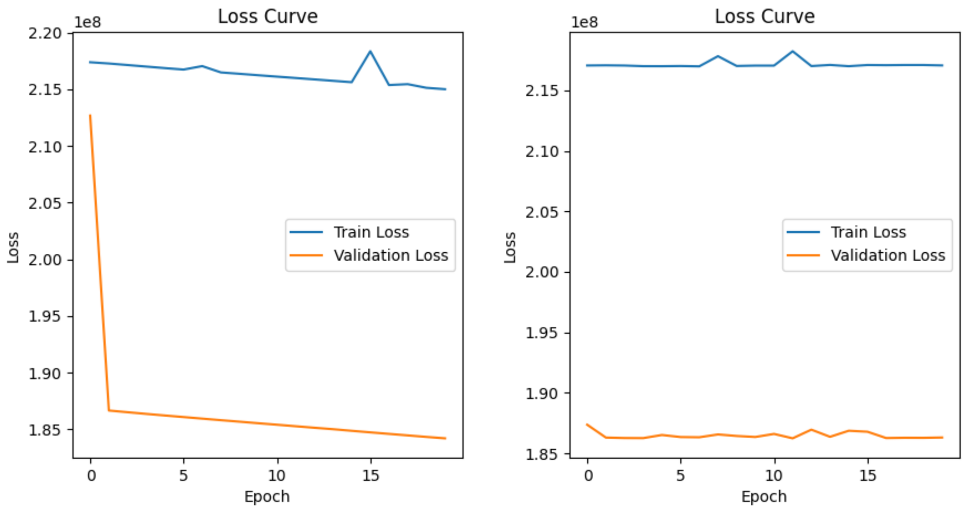
Right image: Learning rate = 1

Dataset: 50% consists of operations with three two-digit numbers, and 50% consists of operations with three one-digit numbers.





Dataset: 90% consists of addition and subtraction operations with three numbers, and 10% consists of addition, subtraction, multiplication, and division operations with three numbers.

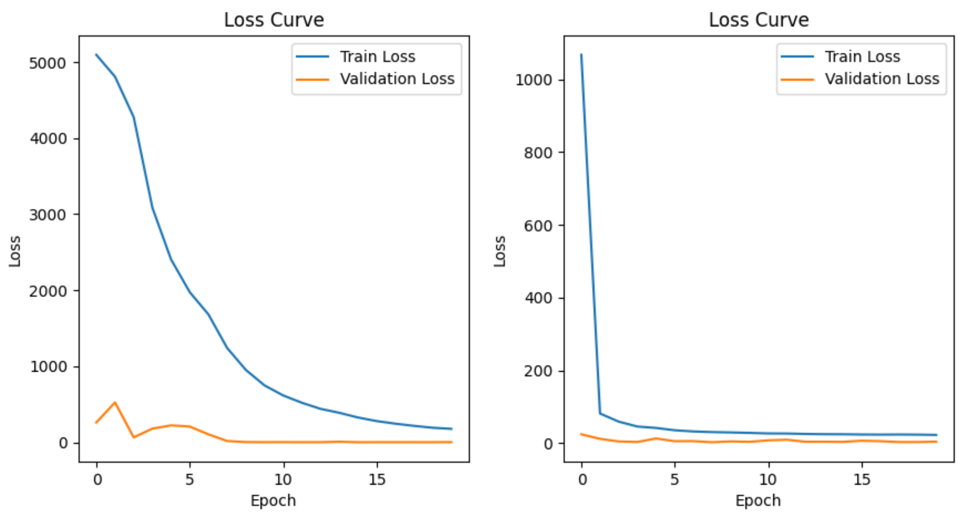


A larger batch size typically results in a smaller range of learning rates for convergence, and the loss tends to be higher. In contrast, a smaller batch size can successfully converge over a wider range of base learning rates, indicating that a smaller batch size allows for a larger range of learning rates for model convergence, making it easier to achieve convergence.

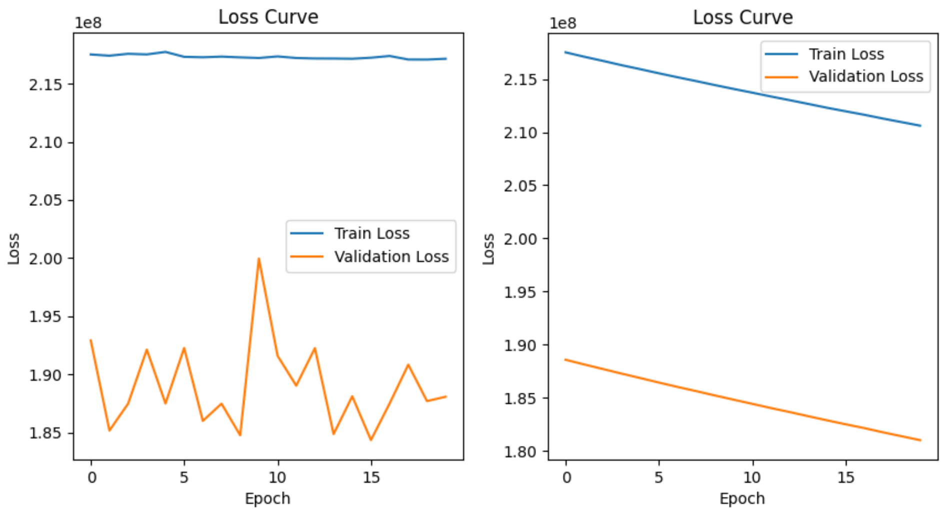
Left image: Batch size = 1024

Right image: Batch size = 16

Dataset: 50% consists of operations with three two-digit numbers, and 50% consists of operations with three one-digit numbers



Dataset: 90% consists of addition and subtraction operations with three numbers, and 10% consists of addition, subtraction, multiplication, and division operations with three numbers.



The model I used is a single-layer RNN. RNNs are particularly well-suited for tasks with sequential characteristics, and since this task only requires predicting the addition and subtraction results of three two-digit numbers, a single-layer RNN is capable of effectively handling this task.

4. I compared the model loss differences among LSTM, RNN, and GRU under the same conditions of batch size, learning rate, and dataset. Since LSTM and GRU have gates that control the filtering of historical data, they are better at preserving long-term memory compared to RNN, which lacks such gates.

Additionally, LSTM has a long-term memory cell that GRU does not have, allowing LSTM to achieve better prediction results than GRU when dealing with long sequences of data.

Below are the training loss and test results.

Left image: LSTM_train_loss

Center image: GRU_train_loss

Right image: RNN_train_loss

```
gpu
Epoch 1/20,Train Loss: 2485.3331619609485
Validation Loss: 892.1834228515625
Epoch 2/20,Train Loss: 542.4871926741166
Validation Loss: 276.97413177490233
Epoch 3/20,Train Loss: 206.62062874707308
Validation Loss: 128.74765074849128
Epoch 4/20,Train Loss: 106.42776262489232
Validation Loss: 76.51806679964065
Epoch 5/20,Train Loss: 64.46848702972585
Validation Loss: 48.55317265689373
Epoch 6/20,Train Loss: 39.64519886062904
Validation Loss: 30.760559445619585
Epoch 7/20,Train Loss: 25.287121736190535
Validation Loss: 21.5284721031785
Epoch 8/20,Train Loss: 17.8411718683148
Validation Loss: 15.26335741877556
Epoch 9/20,Train Loss: 11.808214252983982
Validation Loss: 10.820647180825471
Epoch 10/20,Train Loss: 8.762981032614004
Validation Loss: 8.046158029139042
Epoch 11/20,Train Loss: 6.487427152862603
Validation Loss: 6.930662366747856
Epoch 12/20,Train Loss: 5.4244674779474735
Validation Loss: 5.61372199356556
Epoch 13/20,Train Loss: 4.086877065795389
Validation Loss: 4.2760583207011225
Epoch 14/20,Train Loss: 3.7131903201172296
Validation Loss: 3.9614970952272417
Epoch 15/20,Train Loss: 3.2816888495263727
Validation Loss: 4.144448909163475
Epoch 16/20,Train Loss: 2.646865100980821
Validation Loss: 3.3800942674279213
Epoch 17/20,Train Loss: 2.232427727642723
Validation Loss: 2.563663986325264
Epoch 18/20,Train Loss: 1.8172687644308263
Validation Loss: 1.8884881526231765
Epoch 19/20,Train Loss: 1.5285176970911296
Validation Loss: 1.6264839958399535
Epoch 20/20,Train Loss: 1.1268559418280015
Validation Loss: 1.2997114099562168
```

```
gpu
Epoch 1/20,Train Loss: 2334.924268289046
Validation Loss: 824.3733024597168
Epoch 2/20,Train Loss: 514.6542138186368
Validation Loss: 260.72221126556394
Epoch 3/20,Train Loss: 191.77358598939398
Validation Loss: 111.4507495611906
Epoch 4/20,Train Loss: 89.67856639759107
Validation Loss: 57.614952969551084
Epoch 5/20,Train Loss: 48.004060966047376
Validation Loss: 33.07408984899521
Epoch 6/20,Train Loss: 28.35907982154326
Validation Loss: 20.40225870013237
Epoch 7/20,Train Loss: 17.140618570318278
Validation Loss: 13.347754579782485
Epoch 8/20,Train Loss: 10.881443574262613
Validation Loss: 8.730568542331458
Epoch 9/20,Train Loss: 8.222242123870688
Validation Loss: 6.375298416614532
Epoch 10/20,Train Loss: 5.730487226000563
Validation Loss: 5.202543215453625
Epoch 11/20,Train Loss: 4.033219378526238
Validation Loss: 3.763177241384983
Epoch 12/20,Train Loss: 3.0595707386224107
Validation Loss: 3.047822338342667
Epoch 13/20,Train Loss: 2.6073757530274717
Validation Loss: 3.134955143928528
Epoch 14/20,Train Loss: 2.078609175133434
Validation Loss: 2.0556666895747187
Epoch 15/20,Train Loss: 1.7484006694602696
Validation Loss: 1.709609030932188
Epoch 16/20,Train Loss: 1.2474573230878874
Validation Loss: 1.9433195888996124
Epoch 17/20,Train Loss: 1.212274820205163
Validation Loss: 1.185354833304882
Epoch 18/20,Train Loss: 1.0838401046666233
Validation Loss: 1.4211115375161172
Epoch 19/20,Train Loss: 1.004225804047151
Validation Loss: 1.2597775414586068
Epoch 20/20,Train Loss: 1.1268335195596924
Validation Loss: 1.1671554818749428
```

```
==================================
gpu
Epoch 1/20,Train Loss: 3120.1543201099744
Validation Loss: 962.1651489257813
Epoch 2/20,Train Loss: 510.47384238243103
Validation Loss: 220.05257592201232
Epoch 3/20,Train Loss: 159.76157944310796
Validation Loss: 86.51458344459533
Epoch 4/20,Train Loss: 71.55634008077058
Validation Loss: 47.46967387199402
Epoch 5/20,Train Loss: 38.39054435762492
Validation Loss: 28.278652346134187
Epoch 6/20,Train Loss: 23.349774977700278
Validation Loss: 16.081708499789237
Epoch 7/20,Train Loss: 14.661707812073557
Validation Loss: 11.499975323677063
Epoch 8/20,Train Loss: 10.991881454194134
Validation Loss: 8.291507011651992
Epoch 9/20,Train Loss: 8.982488203455102
Validation Loss: 6.697455036640167
Epoch 10/20,Train Loss: 6.47734352404421
Validation Loss: 7.928260564804077
Epoch 11/20,Train Loss: 4.724084254184907
Validation Loss: 4.728046950697899
Epoch 12/20,Train Loss: 6.111019644886255
Validation Loss: 9.543247151374818
Epoch 13/20,Train Loss: 5.5352473966777325
Validation Loss: 7.157646048069
Epoch 14/20,Train Loss: 3.6601612838831814
Validation Loss: 2.913381803035736
Epoch 15/20,Train Loss: 3.4509933248839597
Validation Loss: 3.639296770095825
Epoch 16/20,Train Loss: 2.784411586651748
Validation Loss: 5.142516875267029
Epoch 17/20,Train Loss: 3.3094977013428104
Validation Loss: 2.2212479077279568
Epoch 18/20,Train Loss: 3.3382478298788723
Validation Loss: 2.925554132461548
Epoch 19/20,Train Loss: 3.2949212794615463
Validation Loss: 3.98143692612648
Epoch 20/20,Train Loss: 3.374931660565463
Validation Loss: 6.02193198800087
```

We can observe that the loss results for LSTM and GRU are quite similar, both around 1. This is due to the relatively short length of the data. In contrast, RNN performs worse and experiences loss fluctuations during training.

Left image: LSTM_test_loss

Center image: GRU_test_loss

Right image: RNN_test_loss

```
('1-8-4=',)-12
('9-0+3=',)12
('0-0+9=',)9
('8+6+1=',)15
('2-2-9=',)-10
('9+4+9=',)22
('6+9+1=',)16
('0+3-8=',)-5
('5-6+9=',)8
('5+7+6=',)18
('7+4-8=',)3
('9+5+8=',)22
('0+3-9=',)-6
('0+8+2=',)10
('1+8+4=',)13
('4-4+4=',)4
('8+4-4=',)8
('4+3+1=',)8
('9+7-1=',)15
('4-8-2=',)-7
('5+0-9=',)-4
('9-7+5=',)7
('5+5-1=',)9
('9-1+5=',)13
('6-8+5=',)3
('4-7+3=',)0
('7+2-1=',)8
('9-6+0=',)3
('7+9-9=',)7
('0-4+5=',)1
('1-0-5=',)-4
('7-2+6=',)11
('0+3-9=',)-6
('9+7-1=',)15
('3-4+5=',)4
('1+7-8=',)0
('0-5-1=',)-7
('3-4-3=',)-5
('1+3-0=',)4
('2+3-2=',)3
('9+2+0=',)11
('1-1-0=',)0
('1+1+0=',)2
('7-7-9=',)-10
('9+3+5=',)17
('4-8+6=',)2
('3+3-7=',)-1
('2+2-2=',)2
('1+7-9=',)-1
('4-0+3=',)7
('1-5+6=',)2
('0+0-4=',)-4
('8-3-3=',)2
Test Loss: 0.9026635154734538
```

```
('1-8-4=',)-11
('9-0+3=',)12
('0-0+9=',)9
('8+6+1=',)15
('2-2-9=',)-9
('9+4+9=',)22
('6+9+1=',)16
('0+3-8=',)-5
('5-6+9=',)8
('5+7+6=',)18
('7+4-8=',)3
('9+5+8=',)22
('0+3-9=',)-6
('0+8+2=',)10
('1+8+4=',)13
('4-4+4=',)4
('8+4-4=',)8
('4+3+1=',)8
('9+7-1=',)15
('4-8-2=',)-6
('5+0-9=',)-4
('9-7+5=',)7
('5+5-1=',)9
('9-1+5=',)13
('6-8+5=',)3
('4-7+3=',)0
('7+2-1=',)8
('9-6+0=',)3
('7+9-9=',)7
('0-4+5=',)1
('1-0-5=',)-4
('7-2+6=',)11
('0+3-9=',)-6
('9+7-1=',)15
('3-4+5=',)4
('1+7-8=',)0
('0-5-1=',)-6
('3-4-3=',)-4
('1+3-0=',)4
('2+3-2=',)3
('9+2+0=',)11
('1-1-0=',)-1
('1+1+0=',)2
('7-7-9=',)-9
('9+3+5=',)17
('4-8+6=',)2
('3+3-7=',)-1
('2+2-2=',)2
('1+7-9=',)-1
('4-0+3=',)7
('1-5+6=',)2
('0+0-4=',)-4
('8-3-3=',)2
Test Loss: 0.9625463007714389
```

```
('1-8-4=',)-12
('9-0+3=',)12
('0-0+9=',)9
('8+6+1=',)15
('2-2-9=',)-9
('9+4+9=',)21
('6+9+1=',)15
('0+3-8=',)-5
('5-6+9=',)8
('5+7+6=',)18
('7+4-8=',)2
('9+5+8=',)21
('0+3-9=',)-6
('0+8+2=',)10
('1+8+4=',)13
('4-4+4=',)4
('8+4-4=',)8
('4+3+1=',)8
('9+7-1=',)15
('4-8-2=',)-6
('5+0-9=',)-5
('9-7+5=',)8
('5+5-1=',)9
('9-1+5=',)13
('6-8+5=',)3
('4-7+3=',)0
('7+2-1=',)8
('9-6+0=',)3
('7+9-9=',)7
('0-4+5=',)2
('1-0-5=',)-4
('7-2+6=',)11
('0+3-9=',)-6
('9+7-1=',)15
('3-4+5=',)4
('1+7-8=',)0
('0-5-1=',)-6
('3-4-3=',)-4
('1+3-0=',)4
('2+3-2=',)3
('9+2+0=',)11
('1-1-0=',)0
('1+1+0=',)2
('7-7-9=',)-10
('9+3+5=',)16
('4-8+6=',)2
('3+3-7=',)-1
('2+2-2=',)2
('1+7-9=',)-1
('4-0+3=',)7
('1-5+6=',)2
('0+0-4=',)-4
('8-3-3=',)2
Test Loss: 1.3851681714883288
```

During testing, we can see that the results for LSTM and GRU are quite similar, while RNN performs worse.