

Deep Learning - 89687

Ass3 – Part 2

The two classes we have chosen for this part are prime numbers, the numbers multiple of 7 and the palindrome.

1. Prime Numbers

- **A description of the language.**

The alphabet is a sequence of numbers from 0 to 9, in a variable-length (1 to n) that forms a prime number.

The positive examples are prime numbers and the negative examples are all the number but not prime.

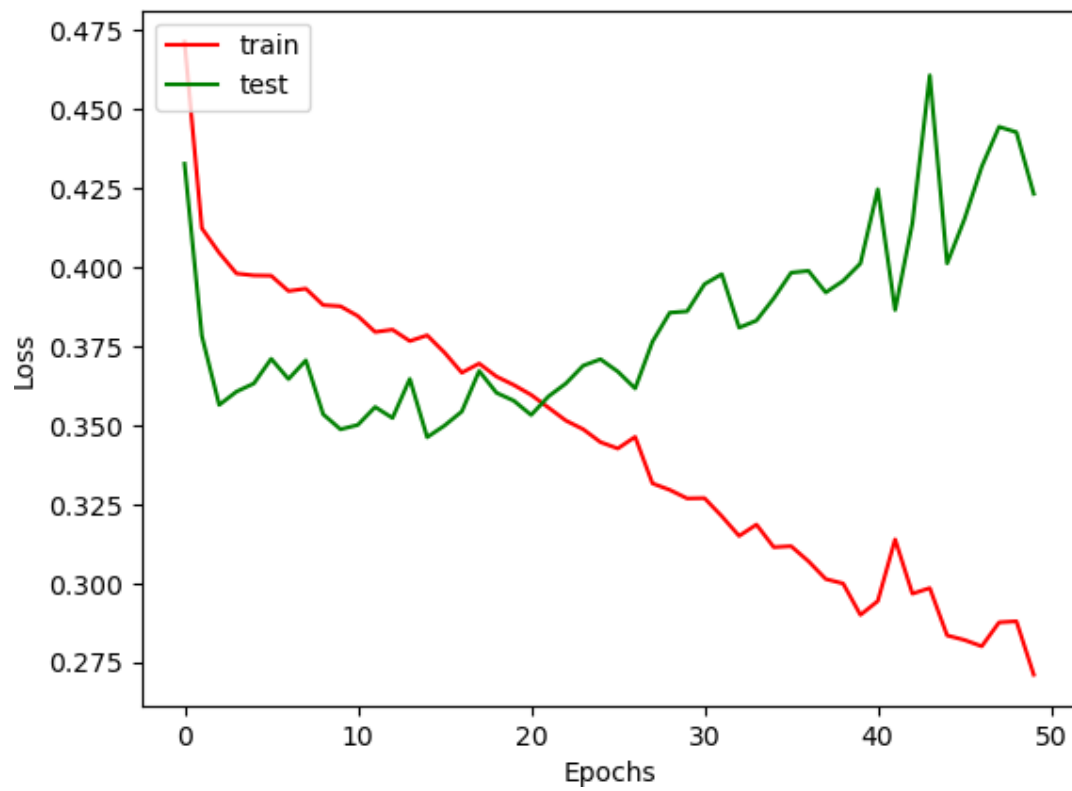
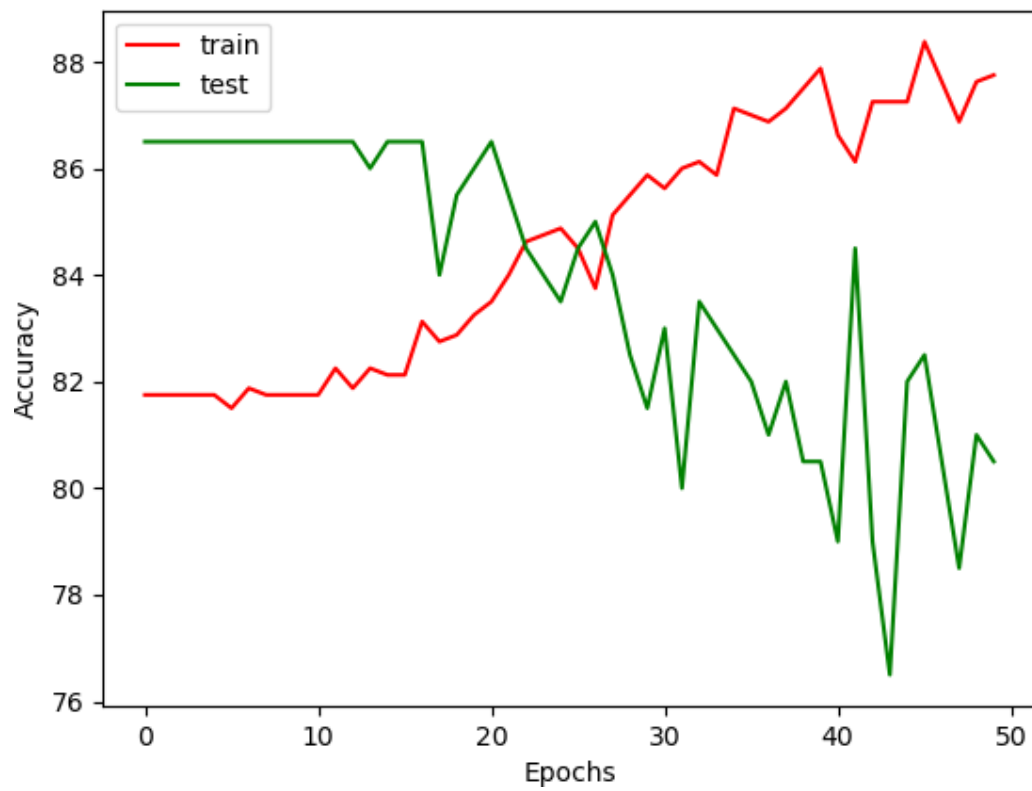
- **Why did you think the language will be hard to distinguish?**

Prime numbers have rules, for example, they don't end by an even number, they are numbers we can divide by one or himself only. We take this into consideration when we created the examples.

- **Did you manage to fail the LSTM acceptor? (including, train and test set sizes, how many iterations did you train for, did it manage to learn the train but did not generalize well to the test, or did it fail also on train?)**

RNNs and LSTM have difficulties learning to generalize a language that has semantical value. Here, we want to learn a sequence of characters that includes a prime number. It is hard to learn a language that is not defined by its sequence of characters, but by the latent meaning of its sequences.

We train the model the same as we did it on part1 with set size of 1000 and 50 epochs. The model learn training set and succeeded with 87.75 on it, but it doesn't find the pattern which will match the validation set also. As we can see on the graphs the loss on test is rising over epoch.



2. Multiple of 7

- **A description of the language.**

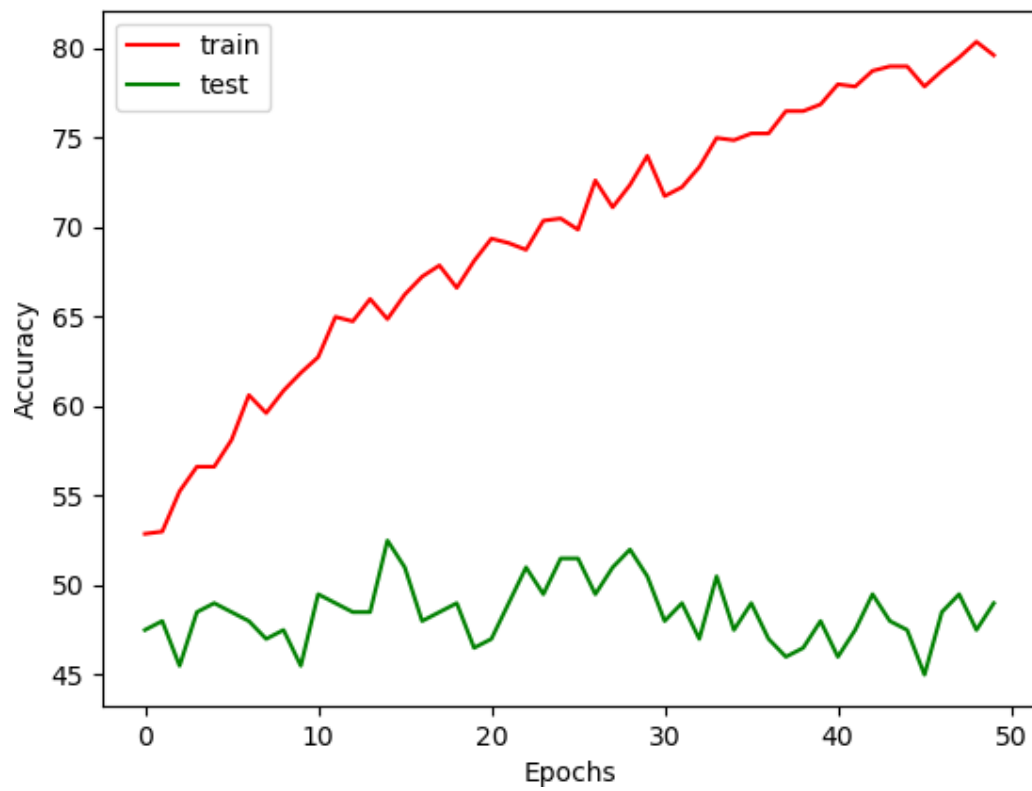
The positive examples are numbers multiple of 7 and the negative examples are all the number but not multiple of 7.

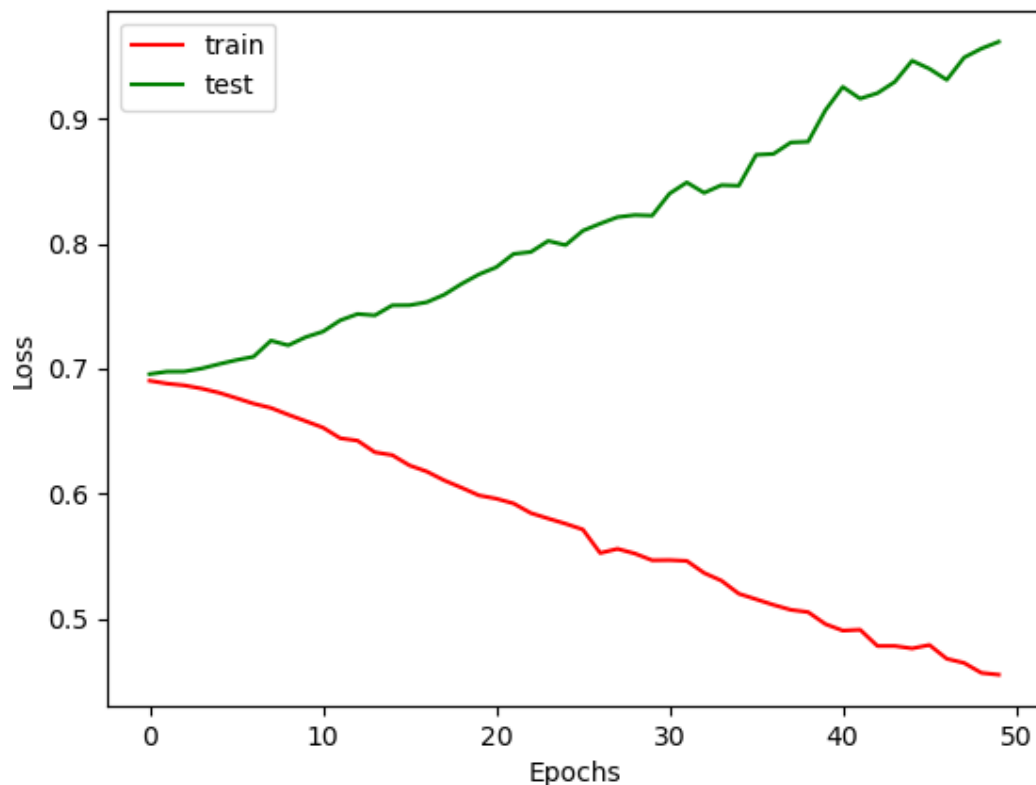
- **Why did you think the language will be hard to distinguish?**

There is no pattern to find multiple of 7.

- **Did you manage to fail the LSTM acceptor? (including, train and test set sizes, how many iterations did you train for, did it manage to learn the train but did not generalize well to the test, or did it fail also on train?)**

We train the model the same as we did it on part1 with set size of 1000 and 50 epochs. The model learn training set and succeeded with 80 on it, but it doesn't find the pattern which will match the validation set also. As we can see on the graphs the loss on test is rising over epoch.





3. Palindrome

- **A description of the language.**

The language is all sequences that read the same backward as forward. We've taken sequence of random numberq, and for each sequence, we have concatenated a reversed sequence to it (e.g. '12345' becomes '123454321').

- **Why did you think the language will be hard to distinguish?**

The LSTM will not remember the order of the first part during the processing of the second part.

- **Did you manage to fail the LSTM acceptor? (including, train and test set sizes, how many iterations did you train for, did it manage to learn the train but did not generalize well to the test, or did it fail also on train?)**

We train the model the same as we did it on part1 with set size of 1000 and 50 epochs. The model learn training set and succeeded with 85% on it, but it doesn't find the pattern which will match the validation set also (50%). As we can see on the graphs the loss on test is rising over epoch.

