DL4Seq - Assignment 3

Otmazgin, Shon
shon711@gmail.com

Rassin, Royi
isroei5700@gmail.com

May 15th, 2021

1   Report 2

Q: A description of the languages
  A:

1. Palindrome language - Alphabet: $\{0, 1\}$. A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam or racecar.

2. Divided by 3 - Alphabet: $\{1\text{-}9\}$, any number divided by 3.

3. Primes language - Alphabet: $\{1\text{-}9\}$, any number which has just two factors: itself and 1.

Q: Why did you think the language will be hard to distinguish?
A:

1. Palindrome language - Beside counting 0s and 1s the model will need to "remember" the location of each character and its mirror, with diffrent length it seems to us hard to distinguish.

2. Divided by 3 - With different size of input lengths We think the model will hard to finding the right neurons that affects the outcome with if number can be divided by 3 any number can be followed by any other number.

3. Primes language - Prime numbers does not have pattern to follow. But the model can distinguish between even and odd numbers and then it should have atleast 50% success rate

Q: 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?)
  A:

1. Palindrome language - Train size: 2100 examples, 1060 palindromes and 1040 non-palindromes. Test size: 900 examples, 440 palindromes and 460 non-palindromes. 1050 iterations with batch size of 10 examples in each iteration. Both train and test reached 50-52% accuracy it always predicted the same label either 0 or 1. The train loss didn't go down after after 10 iterations and remained noisy . The model failed to distinguish between palindromes and non-palindromes numbers. see figure 1-3 for more details.

2. Divided by 3 - Train size: 3500 examples, 1731 examples divided by 3 and 1769 examples not divided by 3. Test size: 1500 examples, 769 examples divided by 3 and 731 examples not divided by 3. 1750 iterations with batch size of 10 examples in each iteration. Both train and test reached 55% accuracy. The train loss remained noisy and the model wasn't able to distinguish between examples divided by 3 or not. see figures 4-6 for more details.

3. Primes language - Train size: 7190 examples, 3583 primes and 3607 non-primes. Test size: 3082 examples, 1553 primes and 1529 non-primes. 3595 iterations with batch size of 10 examples in each iteration. Both train and test reached 83% accuracy. The train loss didn't go down after after 20 iterations and it seems like the model reached his limit. and the model failed to distinguish between primes and non-primes numbers. see figures 7-9 for more details.
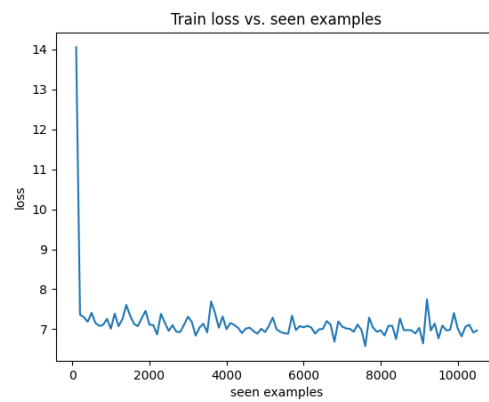
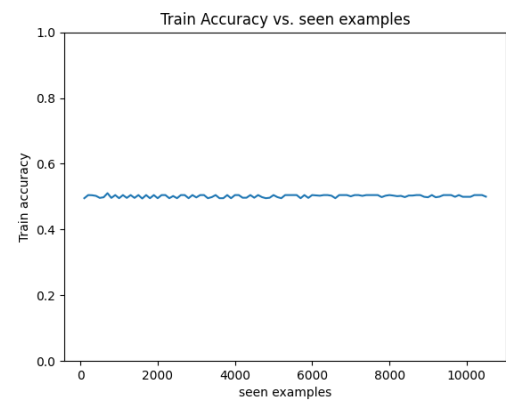Figure 1: Palindromes Train loss

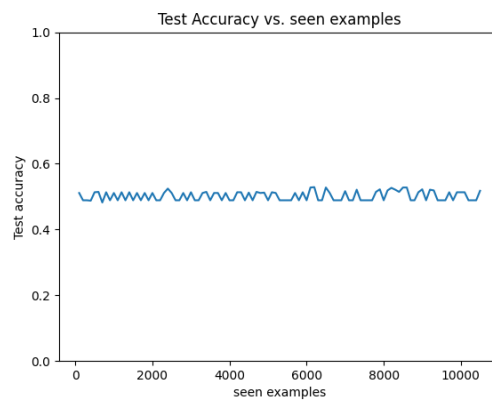Figure 2: Palindromes Train accuracy
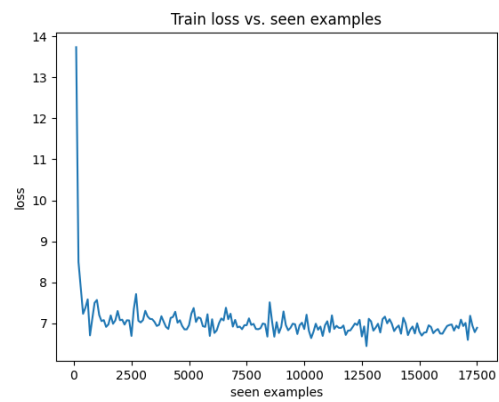

Figure 3: Palindromes test accuracy
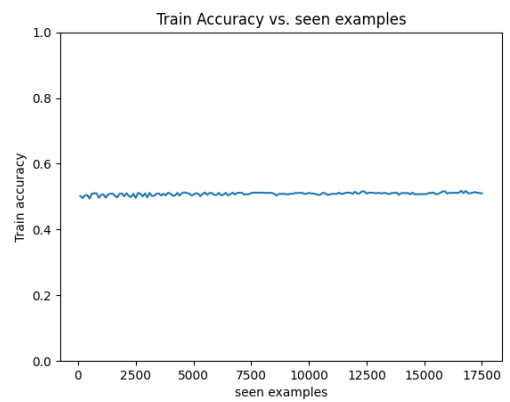
Figure 4: Divided by 3 train loss

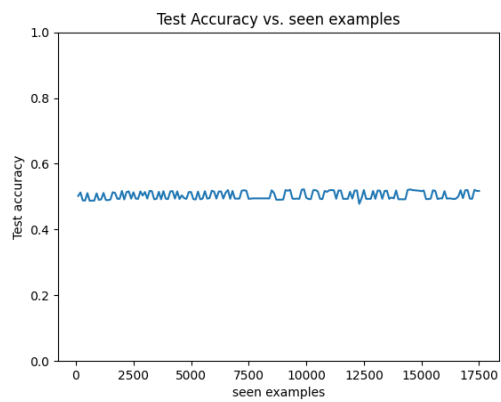Figure 5: Divided by 3 train accuracy

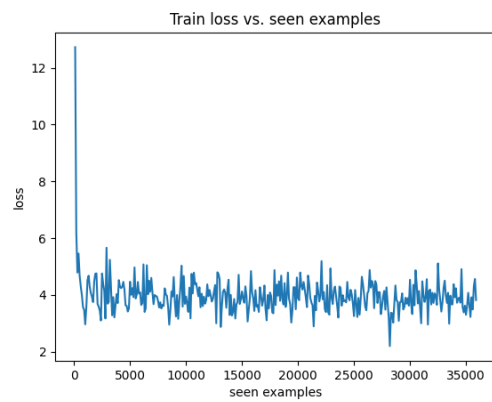
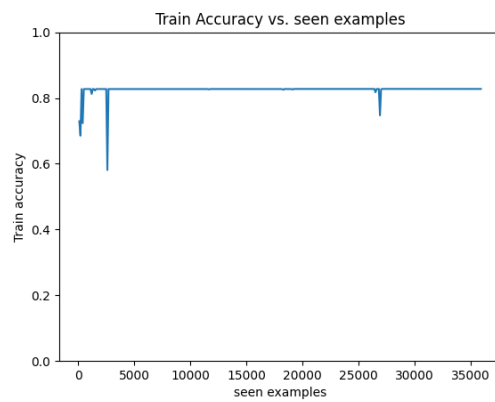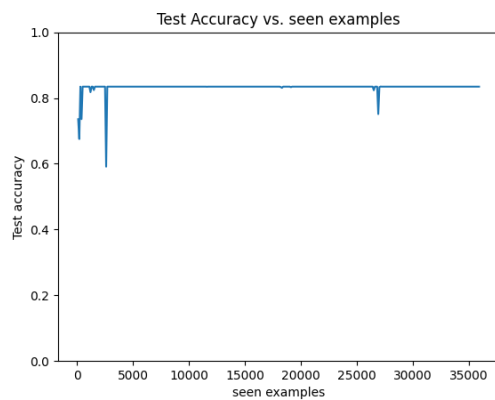Figure 6: Divided by 3 test accuracy

4

Figure 7: Primes train loss

Figure 8: Primes train accuracy

Figure 9: Primes test accuracy