Guy Ben Razon 209207364

Yuval Ramot 208115840

Report2

1. **Description:** Sequences where specific characters or substrings need to match at the beginning and end of the sequence, with a large number of intervening characters.

- **Positive examples:** Let x be a digit (i.e. 0-9) then x[0-9]+x
  - e.g., 12345555555555551, 625615615615616.
- **Negative examples:** Sequences where the beginning and end characters do not match.
  - e.g., 156156123159, 497811561986
- To make it a little harder, all of the words length are of lengths of 10-21.
- **Reason for difficulty:** LSTMs are known to struggle with capturing long-distance dependencies, especially when there are many intervening characters. This makes it hard for the network to remember the initial character and match it with the final character after processing a long sequence.
- **Experiment details:** train test sizes 900, 100 respectively. The LSTM accuracy was poor on the test and train, with accuracy of 0.59 on the test set which is nearly close to a random guess.

2. **Description:** Nested pattern.

- **Positive examples:** Let s be a specific random digits pattern of size 5, then the positive examples are s+
  - e.g., 123451234512345, 54321543215432154321
- **Negative examples:** Let s be a specific random digits pattern of size 5, then the negative examples are [0-9]s
  - e.g., abab, aabbb, aaabb

Reason for difficulty: Nested patterns require the network to understand hierarchical dependencies, which are notoriously difficult for RNNs and LSTMs. The network has to keep track of multiple levels of nesting, which is challenging for the sequential nature of RNNs.

3.