

**CECS 551**  
**Assignment 8**  
**Total: 30 Points**

---

General Instruction

- Submit uncompressed file(s) in the Dropbox folder via Canvas (Not email).
- 

1. Design LSTM network and implement it using Keras library to learn simple arithmetic operations. The objective of the network is estimating result of addition or subtraction of two numbers.

- (a) (5 points) Implement a function to generate all pairs of query and answer. The query includes two integer numbers (0~99) and the an operation (+ or -), and the answer includes correct results of the queries. You should have  $100 \times 100 \times 2 = 20,000$  pairs of queries and answers. Please note that the lengths of queries and answers are fixed as 5 and 4, respectively.

Data set

Query(X): '0+0 ', '0-0 ', '0+1 ', '0-1 ', ..., '99+99', '99-99'

Answer(Y): '+0 ', '+0 ', '+1 ', '-1 ' ..., '+198', '+0 '

- (b) (5 points) Implement a function to encode a string into one-hot-encoding scheme. Please note that the dimensions of a queries and answers are fixed as  $5 \times 13$  and  $4 \times 13$ , respectively.

alphabet = ['0','1','2','3','4','5','6','7','8','9','+','-',' ']

Encoding exmaple

'4+27 '

```
[[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
 [0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]]
```

'+31 '

```
[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0],
 [0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]]
```

- (c) (5 points) Implement a Encoder-Decoder LSTM network using the following code. Explain how this code implements Encoder-Decoder scheme.

```
model = Sequential()
model.add(LSTM(?, input_shape=(5, 13), return_sequences=False))
model.add(RepeatVector(4))
model.add(LSTM(?, return_sequences=True))
model.add(Dense(13, activation='softmax'))
```

- (d) (5 points) Shuffle the data set, and use 70% samples as the training set, 15% as the validation set, and 15% as the test set. Train the network and tune the hyper-parameters, then report the best test accuracy and its setting.
- (e) (5 points) Reverse the query and answer strings in the data set, then repeat training with the settings of best test accuracy of (d).
- ‘4+27 ’ -> ‘ 72+4’  
‘+31 ’ -> ‘ 13+’
- (f) (5 points) With the setting of the best test(valid) accuracy, draw the chart of valid accuracy vs. epoch for both non-reverse(baseline) and reverse data set as shown in Figure 1. You can set `validataion_data=val_data` on `model.fit()` for this. (Actual numbers might be different.)

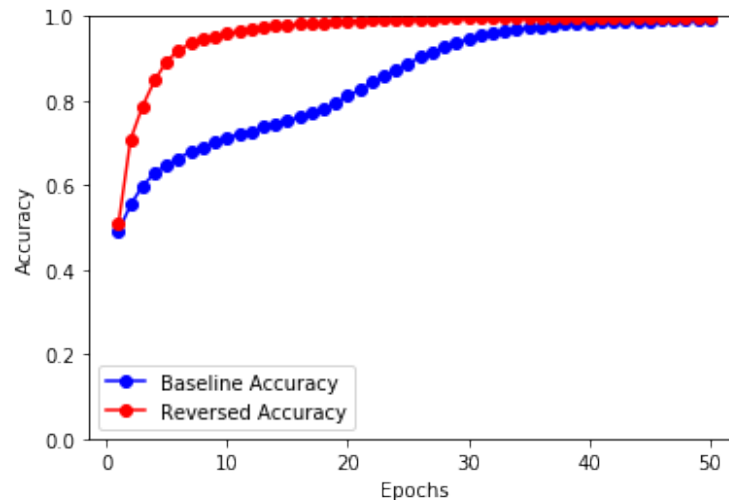


Figure 1: An example of accuracy vs. epoch