# Practical 2. Recurrent Neural Networks and Graph Neural Networks

**Hannah Min**
University of Amsterdam
Student number 11011580
hannah.min@student.uva.nl

## 1 Recurrent Neural Networks

### 1.1 Question

(a)

$$\frac{\partial L^{(T)}}{\partial W_{ph}} = \sum_{i=0}^{t} \frac{\partial L^{(t)}}{y^{(t)}} \frac{y^{(t)}}{\partial p^{(t)}} \frac{\partial p^{(t)}}{\partial W_{ph}}$$

(b)

$$\begin{aligned}\frac{\partial L^{(T)}}{\partial W_{hh}} &= \sum_{i=0}^{t} \frac{\partial L^{(t)}}{y^{(t)}} \frac{y^{(t)}}{\partial p^{(t)}} \frac{\partial p^{(t)}}{\partial h^{(t)}} \frac{\partial h^{(t)}}{\partial h^{(i)}} \frac{\partial h^{(t)}}{\partial W_{hh}} \\ &= \sum_{i=0}^{t} \frac{\partial L^{(t)}}{y^{(t)}} \frac{y^{(t)}}{\partial p^{(t)}} \frac{\partial p^{(t)}}{\partial h^{(t)}} \Big( \prod_{j=i+1}^{t} \frac{\partial h^{(j)}}{\partial h^{(j-1)}} \Big) \frac{\partial h^{(i)}}{\partial W_{hh}}\end{aligned}$$

(c) For $\frac{\partial L^{(T)}}{\partial W_{ph}}$, we only sum over t, and there are no time dependencies within de loop. In contrast, for $\frac{\partial L^{(T)}}{\partial W_{hh}}$, each step in the loop depends on the gradients $\prod_{j=i+1}^{t} \frac{\partial h^{(j)}}{\partial h^{(j-1)}}$, which in each step t requires $T - t$ hidden states from other time steps.

### 1.2 Question

(a) $\mathbf{g}^{(t)}$: This gate combines the new input with the previous hidden state and applies a tanh to it to map its values between -1 and 1. In this way we regulate the size of the values.
$\mathbf{i}^{(t)}$: Input gate: new input and previous hidden state are combined and based on the sigmoid we get an 'importance'. The sigmoid squashes values between 0 and 1, which we can interpret as an importance weight.
$\mathbf{f}^{(t)}$: Forget gate: Based on the output of the sigmoid function, we decide which information to keep and which information to discard.
$\mathbf{o}^{(t)}$: The output gate is the first step in determining what the next hidden state is going to be. It combines input en previous hidden into a sigmoid, which is later multiplied with the current cell state.

(b) Number of trainable parameters:

$$N_{parameters} = 4 \cdot N_{hidden} \cdot N_{input} + 4 \cdot N_{hidden} \cdot N_{hidden} + 4 \cdot N_{hidden} + N_{output} \cdot N_{hidden} + N_{output}$$

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

### 1.3 Question

Model: LSTM Dataset: random combinations
3 random seeds
T: 5, 10, 15

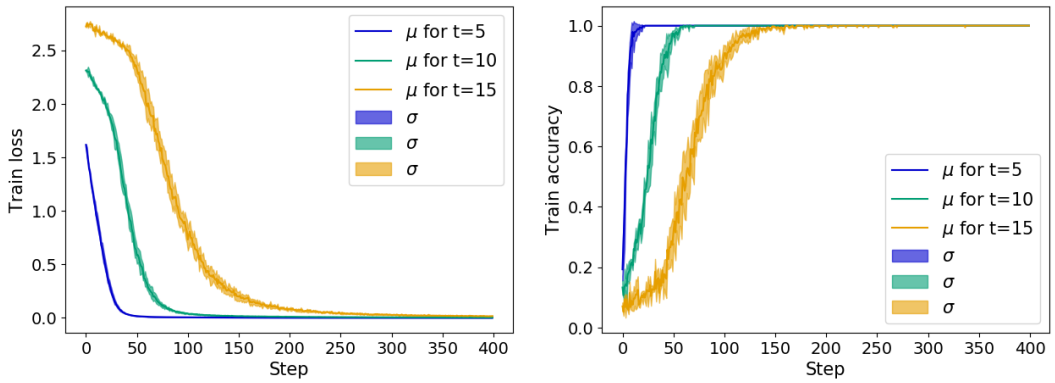


Figure 1: Performance of LSTM model on the random combinations dataset. Scores are averaged over 3 runs.

As this is a very simple task, the model reaches perfect accuracy each time. An LSTM can reach perfect accuracy, because it is able to store the information of the sequence and use this to predict. In our data, there is no information missing to make a perfect prediction. The longer the sequence, the longer it takes to reach this optimum, as we see in figure 1 that it takes longest for t=15.

### 1.4 Question

Model: Bidirectional LSTM
Dataset: random combinations
3 random seeds
Sequence length: 5, 10, 15

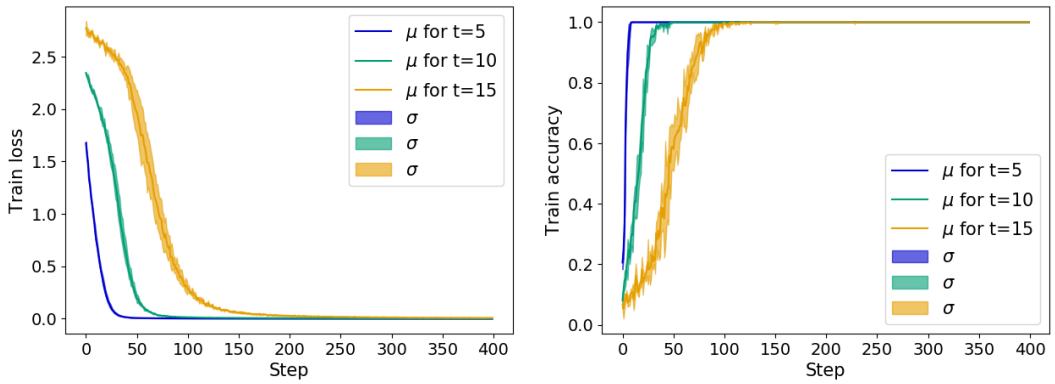


Figure 2: Performance of bidirectional LSTM model on the random combinations dataset. Scores are averaged over 3 runs.

The results of the bidirecional LSTM are very similar to basic LSTM. The bidirectional LSTM reaches perfect accuracy faster, because it is a more complex model, and has more parameters to reach perfect accuracy faster.

## 2 Recurrent Nets as Generative Model

### 2.1 Question

(a) Model: two-layer LSTM
Dataset: Grimms fairy tails

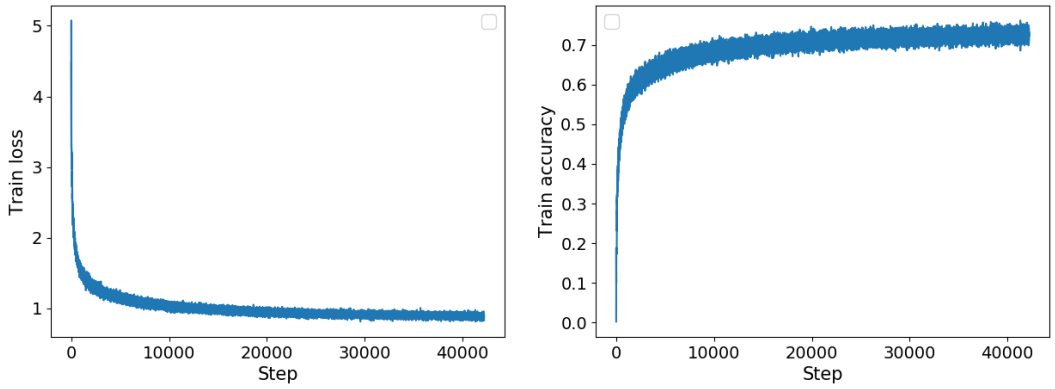


Figure 3: Accuracy and loss curves on training set.

**Hyperparameters:**
I used default hyperparameters, except for the number of training steps. I used $train\_steps = 45000$, because my model converges within this number of steps.

(b) **1/3 of training**
Train Step 16599, Batch Size = 64, Accuracy = 0.70, Loss = 0.995
**sequence length 20:**
1) When the king said t
2) So he said, 'I will
3) Now when the king ha
4) E THE GOOSE-GIRL
5) 1.E.1 The followin

**sequence length 30:**
1) But the second son said, 'I wi
2) and the soldier was already da
3) Chanticleer and mother and sai
4) [EL IN THE BUSH

A farmer had
5) X AND THE SHOEMAKER

There was

**sequence length 60:**
1) f the chart, and see where it grew up the three steps there w
2) 91-0.txt of the bell
rope, he saw the children were sitting
3) !' said he, 'the whole seven day the door with anyone and mo
4) he was to be his father, and said, 'I will give you my wife

5) Now when the king had the spindle which had no could not fin

**2/3 of training**
Train Step 27799 Batch Size = 64, Accuracy = 0.73, Loss = 0.875

**sequence length 20:**
1) one to the table wh
2) Little Red-Cap, and
3) the world were comin
4) , and the stars were
5) And the king said, ‘

**sequence length 30:**
1) He was as the fox had said on 2) and when she was always called 3) ver stood before her, and the 4) 591-0.zip *****
This and all t 5) g and had to draw beer here, t

**sequence length 60:**
1) d the bear with the stairs with the princess, and the soldie
2) 2. If you do not for you?’ ‘Ah,’ she replied, ‘I will give
3) % of the window, and said, ‘I am so homes who were to dispot
4) 8 The little tailor went bride, there thou gangest!
Alas!
5) 8 The little tailor went bride, there thou gangest!
Alas!

**3/3 of training**
Train Step 42099, Batch Size = 64, Accuracy = 0.74, Loss = 0.876

**sequence length 20:**
1) Project Gutenberg-tm
2) ’ said the cook, ‘an
3) Then the fox said, ‘
4) Jorindel said, ‘I wi
5) ked at the house.

T
**sequence length 30:**
1) E THAT THE FOUNDATION, THE BIR
2) 2591-0.t. The straw there, an
3) Queen had a little window at t
4) ch a little dwarf was greatly
5) X

The king said, ‘I will give

**sequence length 60:**
1) You must be a sing the wolf was so steplate, and said, ‘I wi
2) : ‘If that be all,’ answered the little tailor. ‘I don’t lik
3) ” I have not got into the world, and be a mighty draught of
4) 91-0.E.

1.E.6. If an individual Project Gutenberg-tm elect
5) My masters were sitting on a cloak from the top of the sea,c

My sentences at 1/3 of training are already really good, because I already achieved an acuuracy of 0.70. It would have been more interesting if I did less training steps. Because we do greedy sampling, we obeserve more repetitions. Also, because we train on fairy tails, we often see that someone in the story ’said’ something. Short sentences often contain unfinished words. Longer sentences lose meaning (or are illogical) because only information from 30 characters can be considered at a time.

(c) **Sampling with temperature:**

The temperature parameter $\tau$ can make letters with low probability be more likely to be sampled. The higher $\tau$, the more randomness we introduce. Compared to greedy sampling, we see less repetitions, but when we introduce too much randomness ($\tau = 2$), the sentences are not English and not readible anymore.

$\tau = 0.5$
1) REE LICHTANT WITH NO HOW CONTR
2) Foundation, the table was stil
3) Now hap in the evening the sol
4) s well as he could.
'Will you 5) X AND THE SAUSLAANS
For W

$\tau = 1.0$
1) ProLievious, and could not tak
2) Rose-rol, however, was not unl
3) Sek0up onports. Just
listen he
4) in the garden to me. If you
w
5) k light whatever they could; a

$\tau = 2.0$
1) quepsmin?Cle. She
OoIWf I weE
2) :KenN3) LBART
RIDN'EDsO-
@proardoo, cr
4) t
s3"eined,lyKinyJu, wiYe?[Rf
5) of hb@rsip, my niVed ymur
thTg

# 3 Graph Neural Networks

## 3.1 Question

(a) GCN layer exploits structure by having a representation of connected nodes. Nodes send each other information in the form of vectors. Information from nodes that are close together and connected will reach each other faster. Information will be passed through paths through the network.

(b) For a small number of nodes, we can use the adjacency in the GCN layer, however, this is problematic if we have a large number of nodes (N), because we need tot store a NxN matrix in memory. This can be solved by using a list of connections instead. Another limitation is that if we take the mean over messages, the nodes do not know which information belongs to which node. A possible solution is to use separate weights for the self-connections of the nodes.

## 3.2 Question

(a) Adjacency matrix

$$\tilde{\mathbf{A}} = \mathbf{A} + I_N = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

(b) Sending information from node C to node E takes 4 updates as it has to travel (D, B, A, E) or (D, F, A, E).

### 3.3 Question

### 3.4 Question

2 applications of GNN
- A network can be trained on molecules. New molecules can be generated and the network makes sure that they are chemically valid.
- Relations between people can be modeled with social networks. The network can suggest new connections.

### 3.5 Question

(a)

(b)