Learning Sequences Using Recurrent Neural Networks

Mert Yaşin, Prof. Ethem Alpaydın

Department of Computer Engineering, Boğaziçi University, İstanbul, Turkey

Introduction

- Songs, poems, books, movies, dialogues are just some examples of data found in a **sequential** form. Learning these sequences, more importantly, being able to **generate** these sequences is a difficult task.
- **Recurrent neural network** is a model which can create a dynamic internal state that withholds information about the data.
- **A Long Short-Term Memory neural network** is a more advanced architecture that can remember its 'experience' for a very long time sequence, thus better adapt to the input.

LSTM Model

Long Short-Term Memory neural networks are excellent for learning sequences due to their unique architecture that 'remembers'. A single LSTM cell has a fairly complicated inner structure.

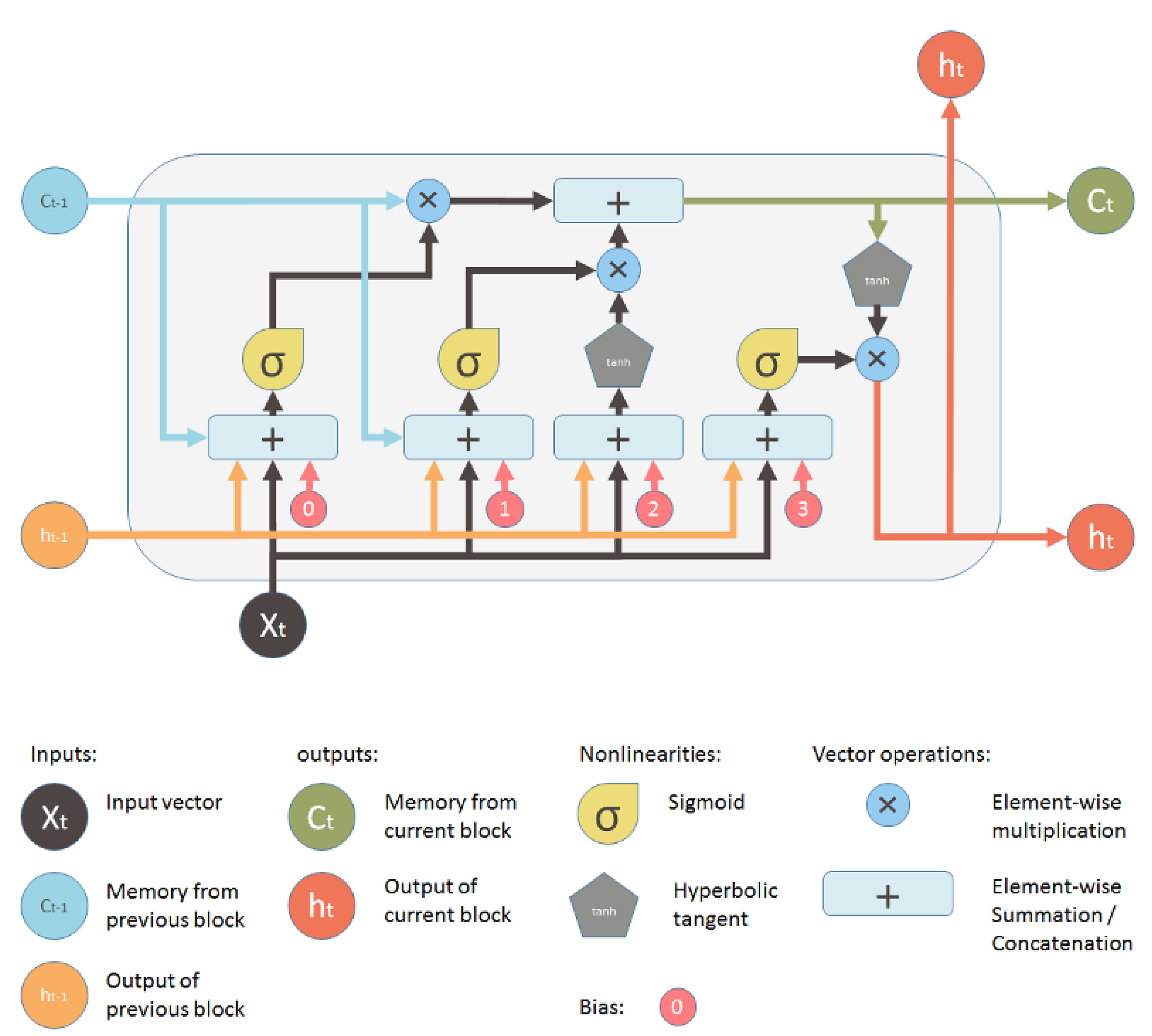

Figure 1: LSTM Cell

LSTM Equations

The equations contain the σ (sigmoid) and and *tanh* (hyperbolic tangent) functions in order to achieve nonlinear neuron-like activations.

$$
\begin{aligned}
z^t &= \tanh\left(W_z x^t + R_z h^{t-1} + b_z\right) && \texttt{block input} \\
i^t &= \sigma\left(W_i x^t + R_i h^{t-1} + b_i\right) && \texttt{input gate} \\
f^t &= \sigma\left(W_f x^t + R_f h^{t-1} + b_f\right) && \texttt{forget gate} \\
c^t &= i^t \odot z^t + f^t \odot c^{t-1} && \texttt{cell state} \\
o^t &= \sigma\left(W_o x^t + R_o h^{t-1} + b_o\right) && \texttt{output gate} \\
h^t &= o^t \odot \tanh\left(c^t\right) && \texttt{block output}
\end{aligned}
$$

The weights W_j are connected to the input layer and the recurrent weights R_j are connected to the previous time instance of the hidden layer. b_j are the bias.

Experiments

Synthetic Data

We created some toy data sets and trained a single **LSTM cell** for each.

01010101010101010101010...	1 zero 1 one
000100010001000100010...	3 zeroes 1 one
000001000001000001000...	5 zeroes 1 one
000000000010000000000...	10 zeroes 1 one
000000000000000000001...	20 zeroes 1 one
000001111100000111110...	5 zeroes 5 ones
000000000011111111110...	10 zeroes 10 ones

Below are some character samplings from the trained networks.

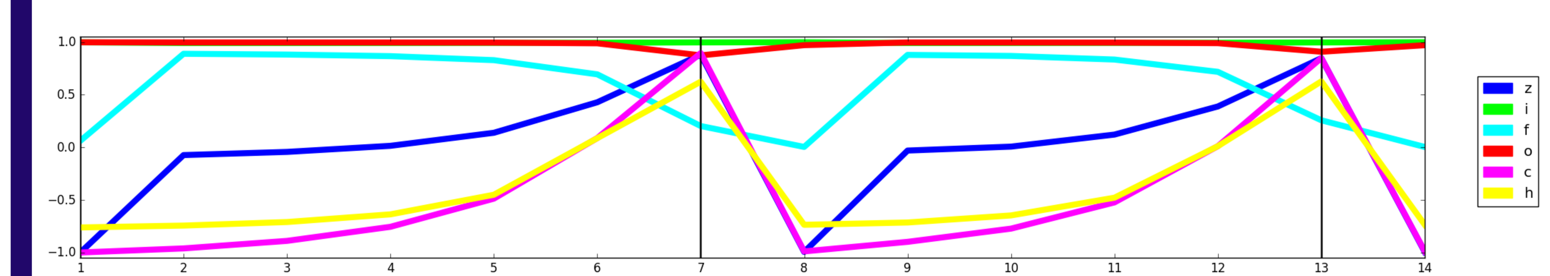

Figure 2: 5 zeroes 1 one - 000001...

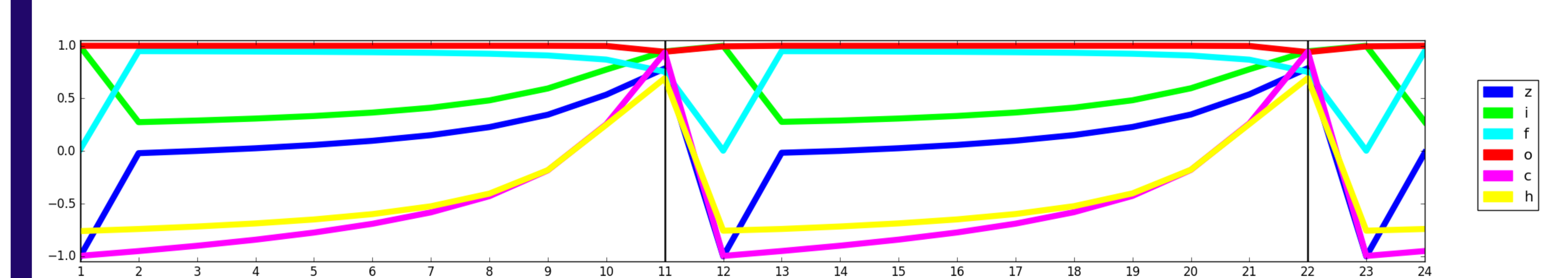

Figure 3: 10 zeroes 1 one - 00000000001...

Although the number of zeroes before every one is different, both of the LSTM cells are able to successfully generate the sequences. This length difference is captured by the small differences in the **weights** of the trained networks. As we can see from the plots, their convergence rate is different. They resemble time stretched versions of one another.

Real Data

GLOUCESTER:
Your good love found dead sword, brokleat;
And with poor bry, that the prosess better a
were alive, 'tis like pardon.

First Watchman:
I cannot lose him. What liege to hear my breathing men,
Is set their exchultic for her wounds wrongs.
God garden, fexor me a heaven,
Have him a man, one anwerged of patricians.

QUEEN ELIZABETH:
With for less shocks a death, after
And did it stay my hand in petet by your brother,
Your fiends thy tumbly at no trask made!

CAPULET:
Go to your heart for you, a Capiso. And, with us.

This short passage has been sampled from a network trained with the plays of the greatest writer in the English language, ***William Shakespeare***.

Experiments

Real Data

Fakat çünkü
Öldü, bir dalda,
Kapıdan
Baktı ele göndereler ne :
"— Aradı şehrini.
Kapılarla yakarım!..
Çocuk, soğukta ne şey?..
–
Kulaklım olduğumu sanmak,
karanlıkta ikinin akşama...
Bir gece aracaktı
Yok!
Sonun,
Yoklaşını
ölü vardır,
seslerin dökülüp etileren
kadar kırın,
sesini.
azımı anlağı gece kalbi geldi.
Geçmiş var :
kızların kapılarındayım."
Durdum...
Bu genç gözlerinden
ve perçeli,
haber gözlerin içini
değil,
Çine, yazmıştı.

Above is a sampled poem from a network that has been trained with the poems of Turkish literary figure, ***Nazım Hikmet***.

The spectacular thing about these results is that the LSTM neural network does not know what words, sentences, new lines, or punctuation means. The only thing the model knows is the characters. It only looks at the sequence of the characters, learns them, and predicts the next one.

Conclusion

- Sequence learning is a problem with a wide range of possible applications.
- Recurrent neural networks are good at learning sequences.
- **LSTM** neural networks are **better** at learning sequences due to their unique representation of long-term dependencies.
- We can learn and generate sequences that are good enough to be mistaken for the original.