LSTM

LSTM: Long Short Term Memory

Long Short-Term Memory is an improved version of recurrent neural network designed by Hochreiter & Schmidhuber. ***LSTM** is well-suited for sequence prediction tasks and excels in capturing long-term dependencies.

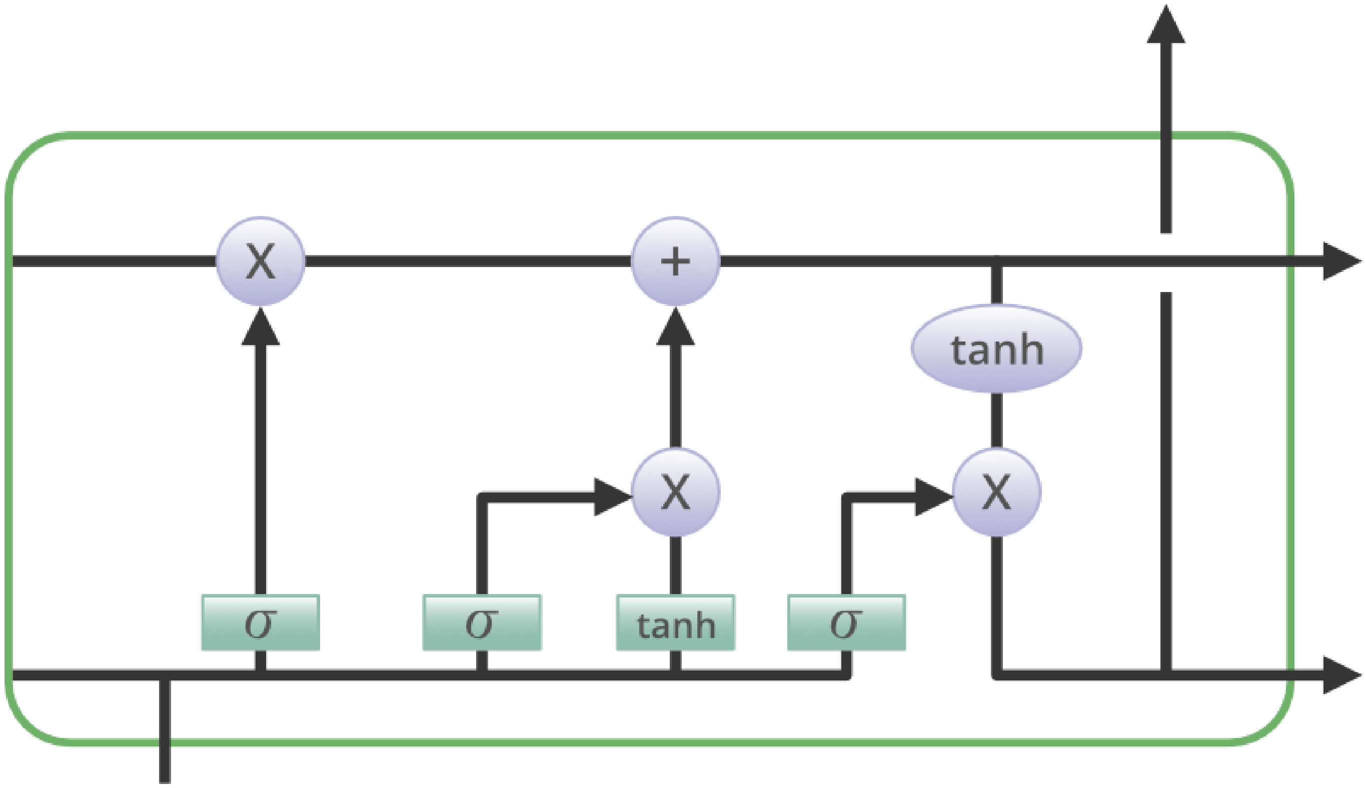
A traditional RNN has a single hidden state that is passed through time, which can make it difficult for the network to learn long-term dependencies. LSTMs address this problem by introducing a memory cell, which is a container that can hold information for an extended period. LSTM networks are capable of learning long-term dependencies in sequential data, which makes them well-suited for tasks such as language translation, speech recognition, and time series forecasting. LSTMs can also be used in combination with other neural network architectures, such as Convolutional Neural Networks (CNNs) for image and video analysis.

Bidirectional LSTM

Bidirectional LSTM (Bi LSTM/ BLSTM) is recurrent neural network (RNN) that is able to process sequential data in both forward and backward directions. This allows Bi LSTM to learn longer-range dependencies in sequential data than traditional LSTMs, which can only process sequential data in one direction.

- Bi LSTMs are made up of two LSTM networks, one that processes the input sequence in the forward direction and one that processes the input sequence in the backward direction. The outputs of the two LSTM networks are then combined to produce the final output.
- Bi LSTM have been shown to achieve state-of-the-art results on a wide variety of tasks, including machine translation, speech recognition, and text summarization.

Architecture and Working of LSTM



Forget Gate

The information that is no longer useful in the cell state is removed with the forget gate. Two inputs x_t (input at the particular time) and h_t-1 (previous cell output) are fed to the gate and multiplied with weight matrices followed by the addition of bias. The resultant is passed through an activation function which gives a binary output. If for a particular cell state the output is 0, the piece of information is forgotten and for output 1, the information is retained for future use. The equation for the forget gate is:

where:

- W_f represents the weight matrix associated with the forget gate.
- [h_t-1, x_t] denotes the concatenation of the current input and the previous hidden state.
- b_f is the bias with the forget gate.
- σ is the sigmoid activation function.

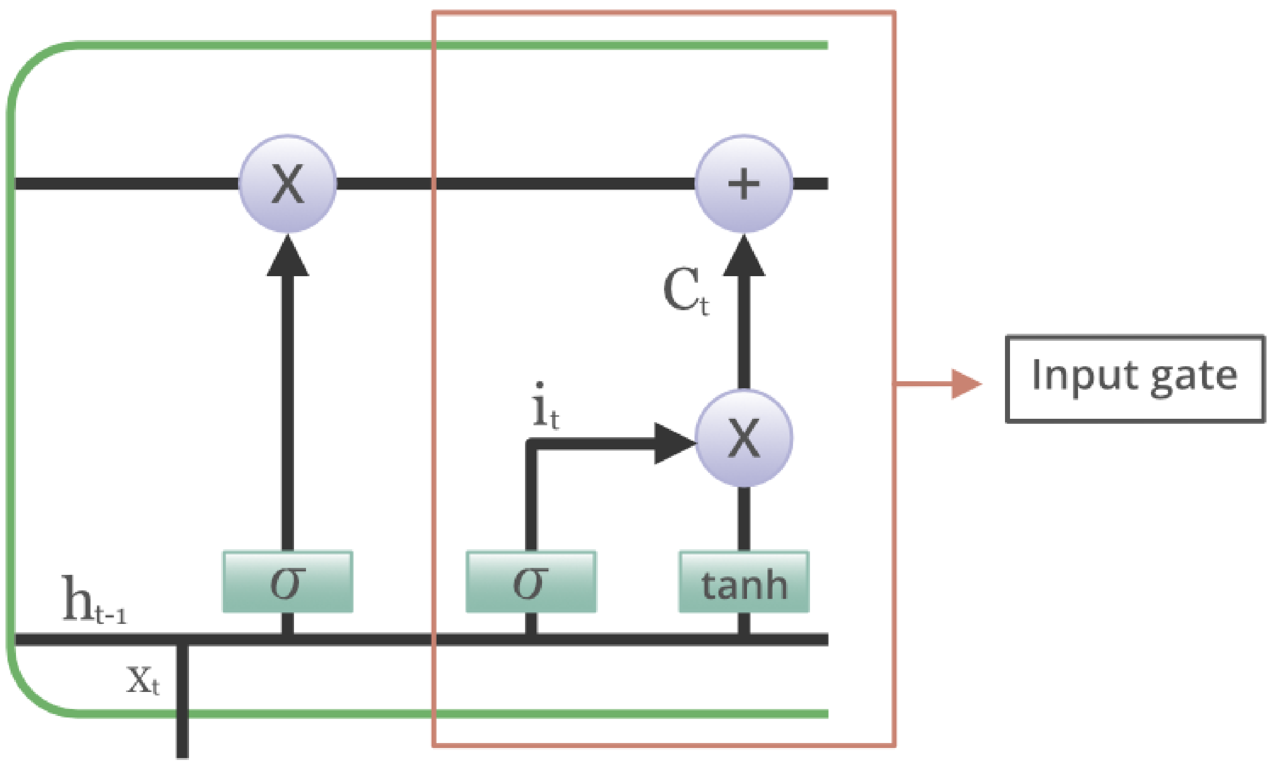
```
![[newContent2 1.jpg]]
```

*Input gate

The addition of useful information to the cell state is done by the input gate. First, the information is regulated using the sigmoid function and filter the values to be remembered similar to the forget gate using inputs h_t-1 and x_t. . Then, a vector is created using **tanh** function that gives an output from -1 to +1, which contains all the possible values from h_t-1 and x_t . At last, the values of the vector and the regulated values are multiplied to obtain the useful information. The equation for the input gate is:

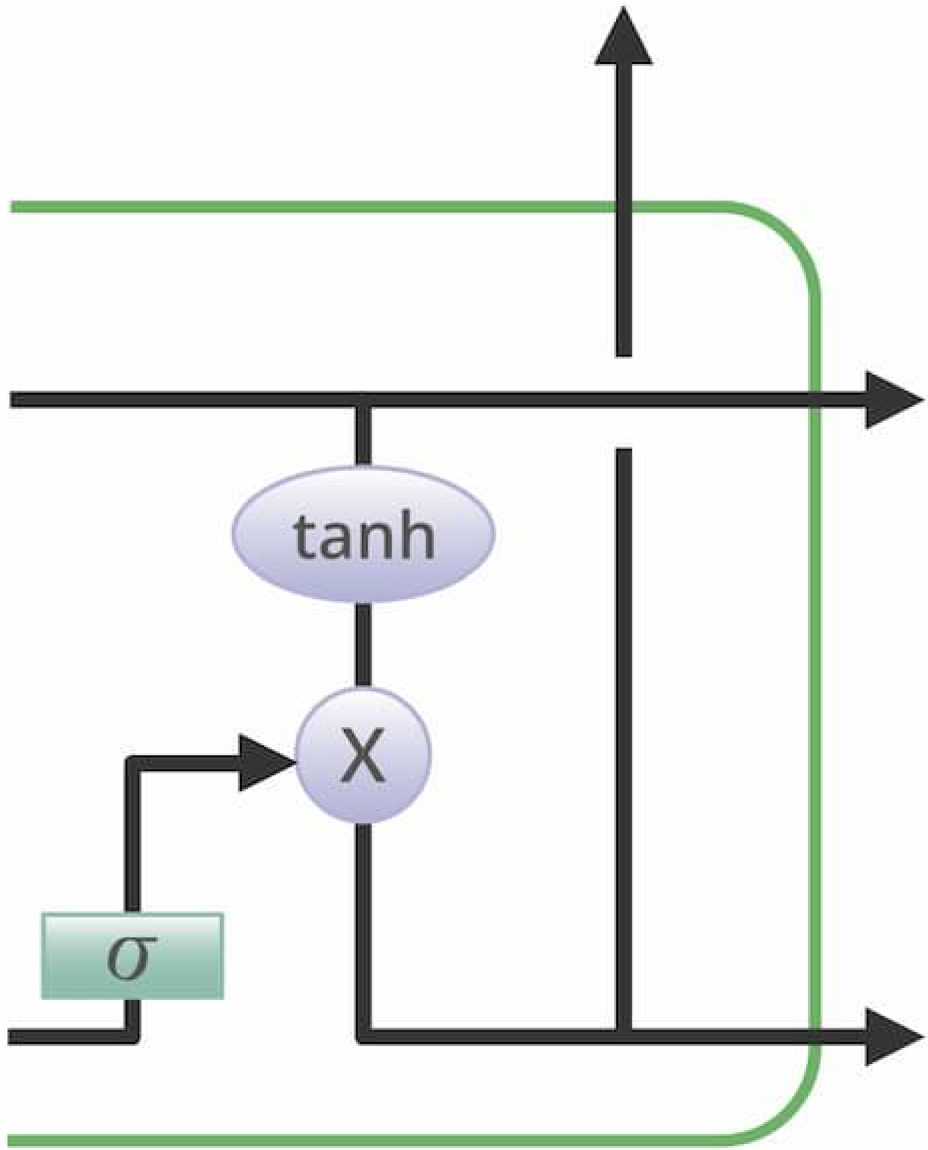
where

- ⊙ denotes element-wise multiplication
- tanh is tanh activation function



Output gate

The task of extracting useful information from the current cell state to be presented as output is done by the output gate. First, a vector is generated by applying tanh function on the cell. Then, the information is regulated using the sigmoid function and filter by the values to be remembered using inputs h_t-1 and x_t. At last, the values of the vector and the regulated values are multiplied to be sent as an output and input to the next cell. The equation for the output gate is:



*Advantages and Disadvantages of LSTM

The advantages of LSTM (Long-Short Term Memory) are as follows:

- Long-term dependencies can be captured by LSTM networks. They have a memory cell that is capable of long-term information storage.
- In traditional RNNs, there is a problem of vanishing and exploding gradients when models are trained over long sequences. By using a gating mechanism that selectively recalls or forgets information, LSTM networks deal with this problem.
- LSTM enables the model to capture and remember the important context, even when there is a significant time gap between relevant events in the sequence. So where understanding context is important, LSTMS are used. eg. machine translation.

The disadvantages of LSTM (Long-Short Term Memory) are as follows:

- Compared to simpler architectures like feed-forward neural networks LSTM networks are computationally more expensive. This can limit their scalability for large-scale datasets or constrained environments.
- Training LSTM networks can be more time-consuming compared to simpler models due to their computational complexity. So training LSTMs often requires more data and longer training times to achieve high performance.
- Since it is processed word by word in a sequential manner, it is hard to parallelize the work of processing the sentences.

LSTM IN PYTHON

TENSOR FLOW
https://www.tensorflow.org/api_docs/python/

KERAS
https://keras.io/api/layers/recurrent_layers/lstm/

MACHINE LEARNING MASTERY
https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/

TOWARDS DATA SCIENCE
https://towardsdatascience.com/exploring-the-lstm-neural-network-model-for-time-series-8b7685aa8cf

Publicação Original sobre LSTM por Hochreiter & Schmidhuber em 1997

http://www.bioinf.jku.at/publications/older/2604.pdf

Fontes

XGEEKS
https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/

Colah Blog
https://colah.github.io/posts/2015-08-Understanding-LSTMs/