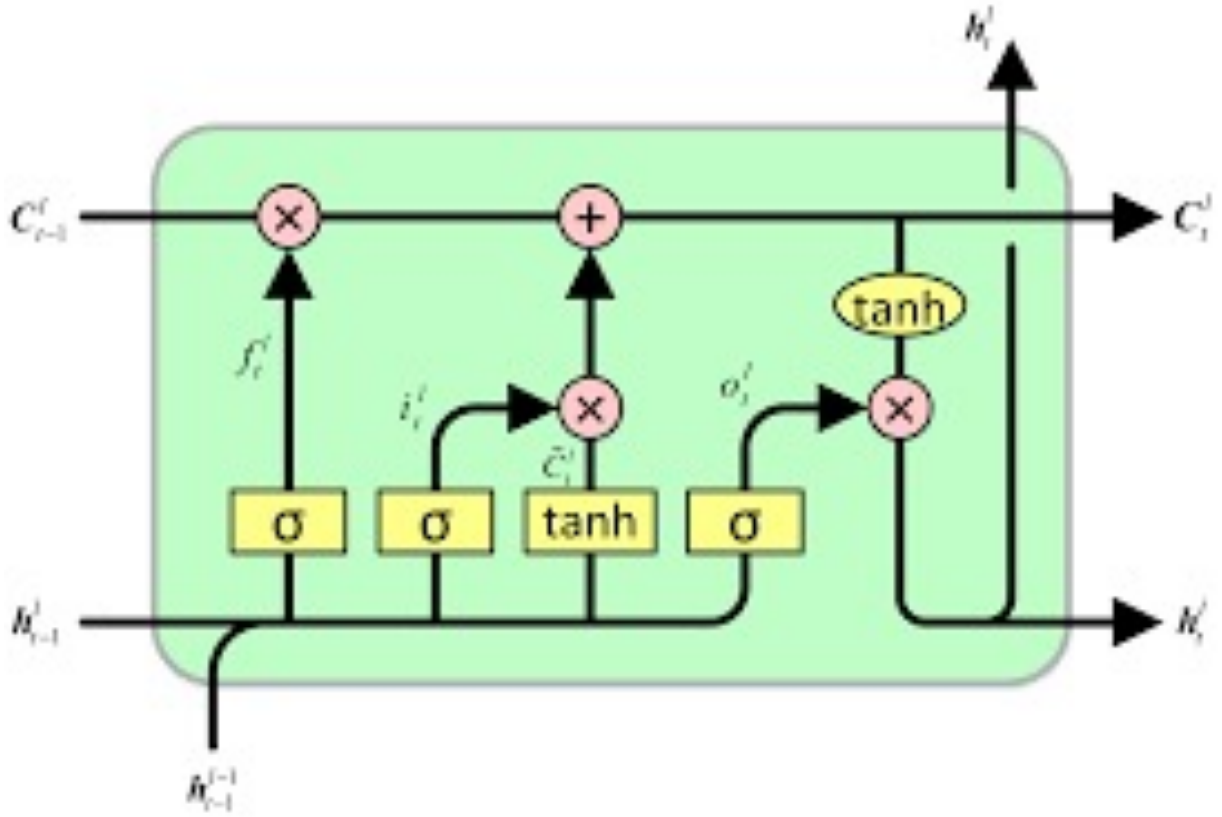


LONG SHORT TERM MEMORY (LSTM)

MR. ASHISH CHOUHAN AND MR. AJINKYA PATIL

SRH HOCHSCHULE HEIDELBERG
Intelligence in Learning

DAS CORE PRINZIP

THE PROBLEM, SHORT-TERM MEMORY

STAATLICH
ANERKANNTE
HOCHSCHULE

- Recurrent Neural Networks suffer from short-term memory.
- If a sequence is long enough, they'll have a hard time carrying information from earlier time steps to later ones.
- If you are trying to process a paragraph of text to do predictions, RNN's may leave out important information from the beginning.

THE PROBLEM OF VANISHING GRADIENT

Why are vanishing gradients a problem?

- In Deep Learning, each of the neural network's weights receive an update proportional to the partial derivate of the error function with respect to the current weight in each iteration of training.
- In some cases, the gradient will be vanishingly small, effectively preventing the weight from changing its value.
- In the worst case, this may completely stop the neural network from further training.

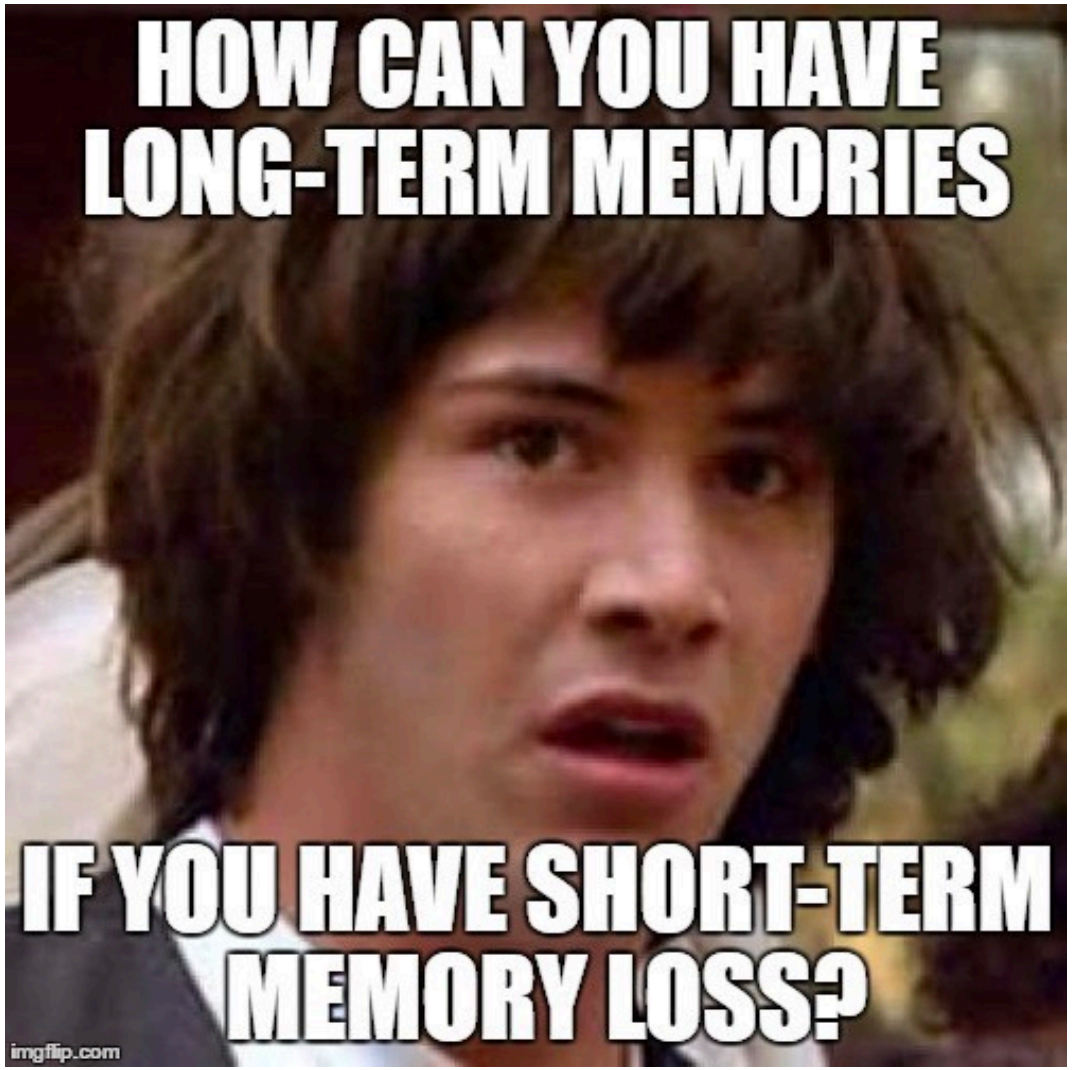
INTRODUCTION TO LSTM

STAATLICH
ANERKANNTE
HOCHSCHULE

- Long-Short-Term Memory(LSTM) models are a type of Recurrent Neural Networks(RNNs) which has the ability to learn and remember over long sequences of input data through the use of **“gates”** which regulate the information flow of the network.

- Long Short Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber (1997),

INTRODUCTION TO LSTM



INTRODUCTION TO LSTM

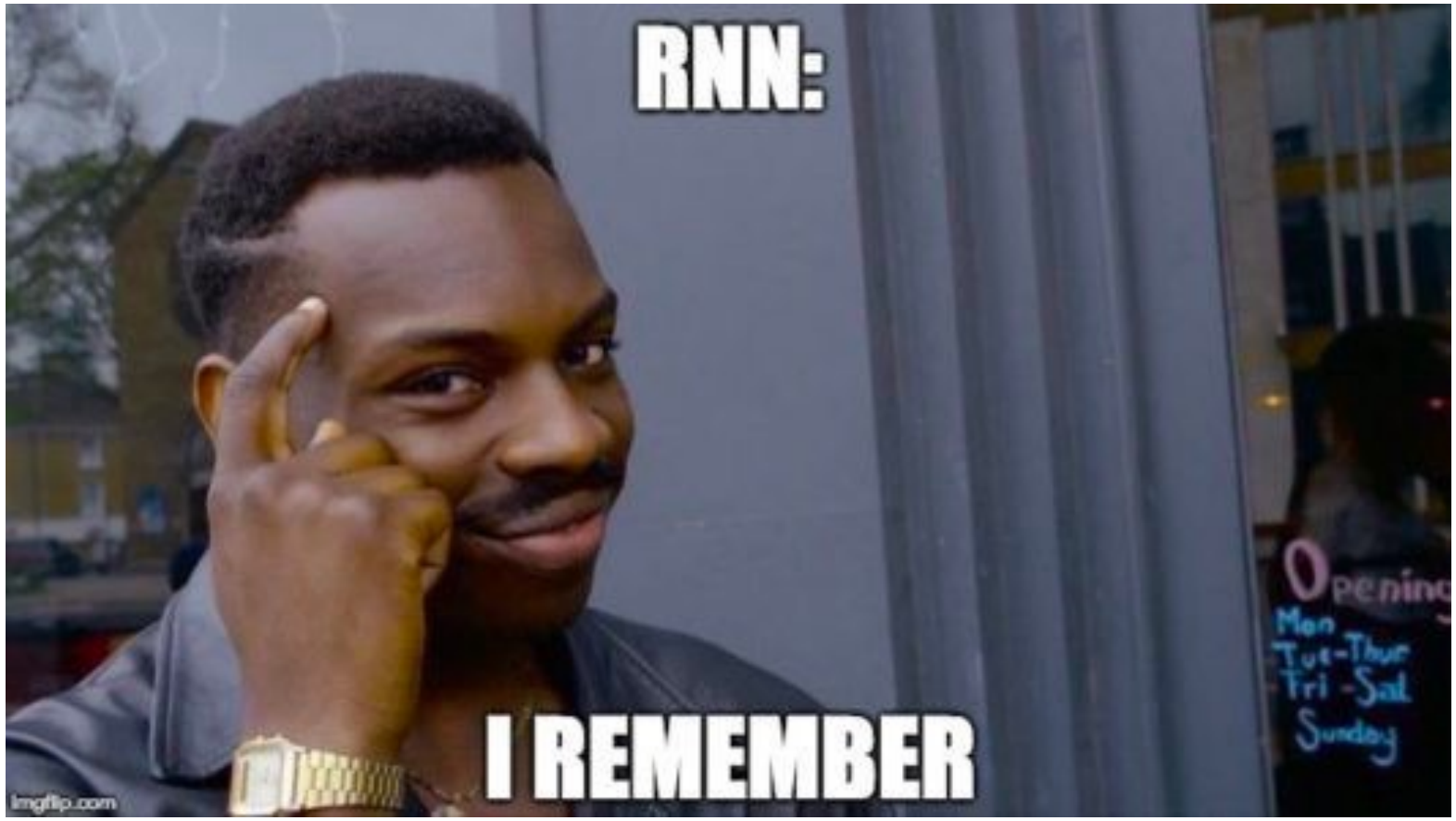
STAATLICH
ANERKANNTE
HOCHSCHULE

LSTMs address the following limitations of RNNs.

- **Short-term memory** — Discarding information from earlier time steps when moving to later ones, which result in the loss of important information.

- **Vanishing gradient** — The gradient is the value used to update the weight used in a neural network. If a gradient value becomes extremely small, it doesn't contribute too much to learning.

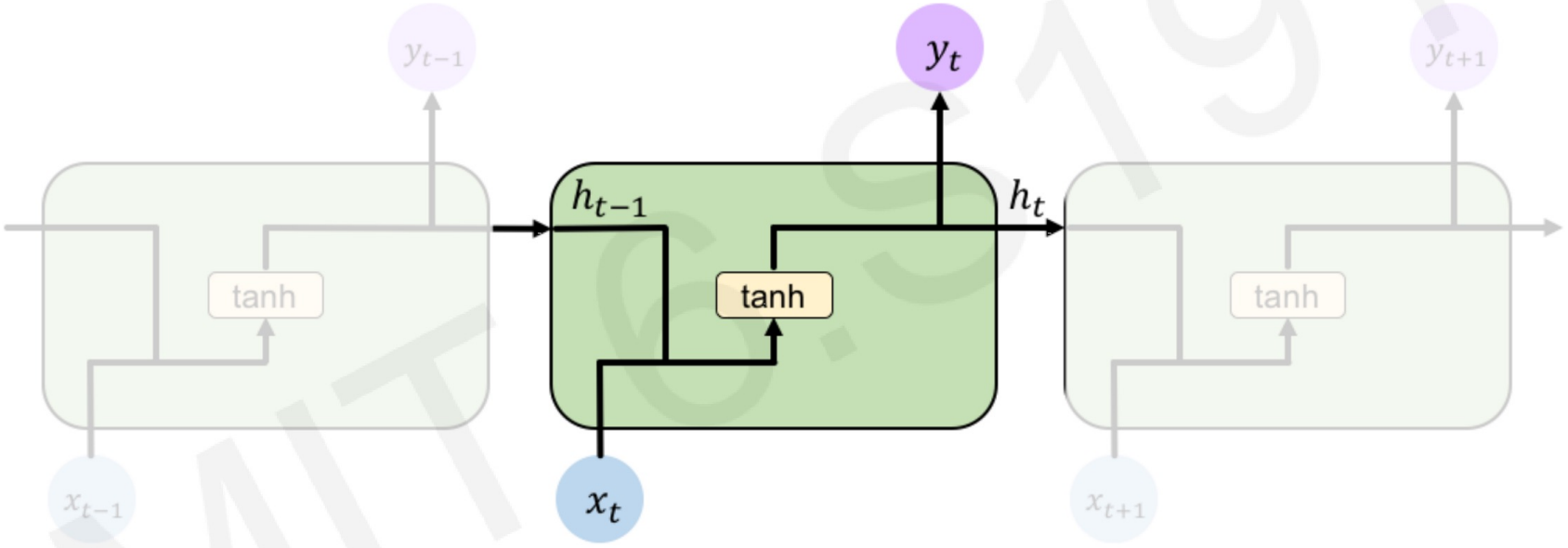
WHAT IS LSTM?



STANDARD RNN

STAATLICH
ANERKANNTE
HOCHSCHULE

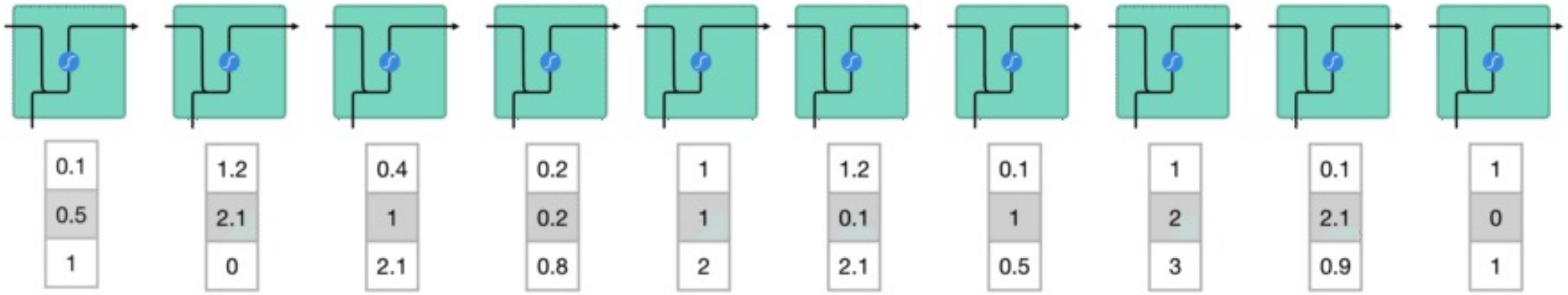
In a standard RNN, repeating modules contain a **simple computation node**.



STANDARD RNN

STAATLICH
ANERKANNTE
HOCHSCHULE

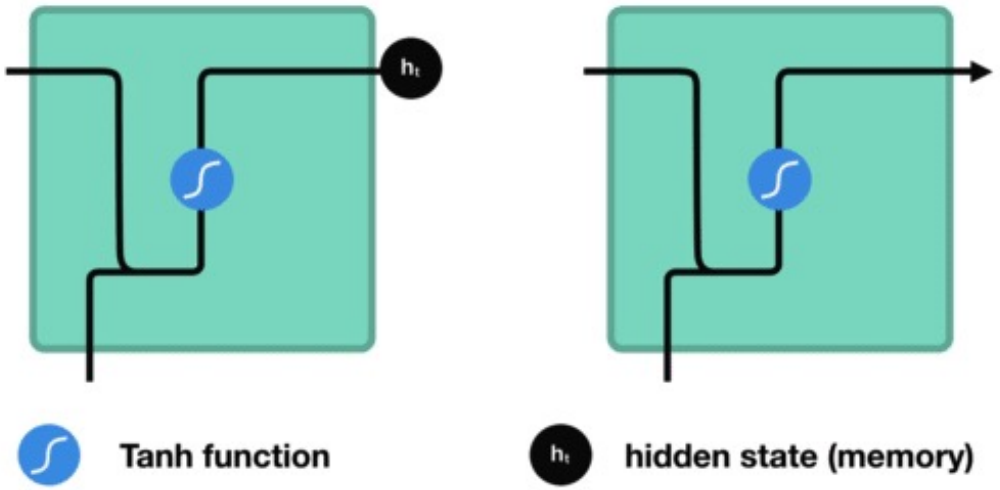
1. Convert data to machine readable vectors.
2. Process vectors one-by-one.



STANDARD RNN

STAATLICH
ANERKANNTE
HOCHSCHULE

1. Passes previous hidden state to next step of sequence.



LSTM - OVERVIEW

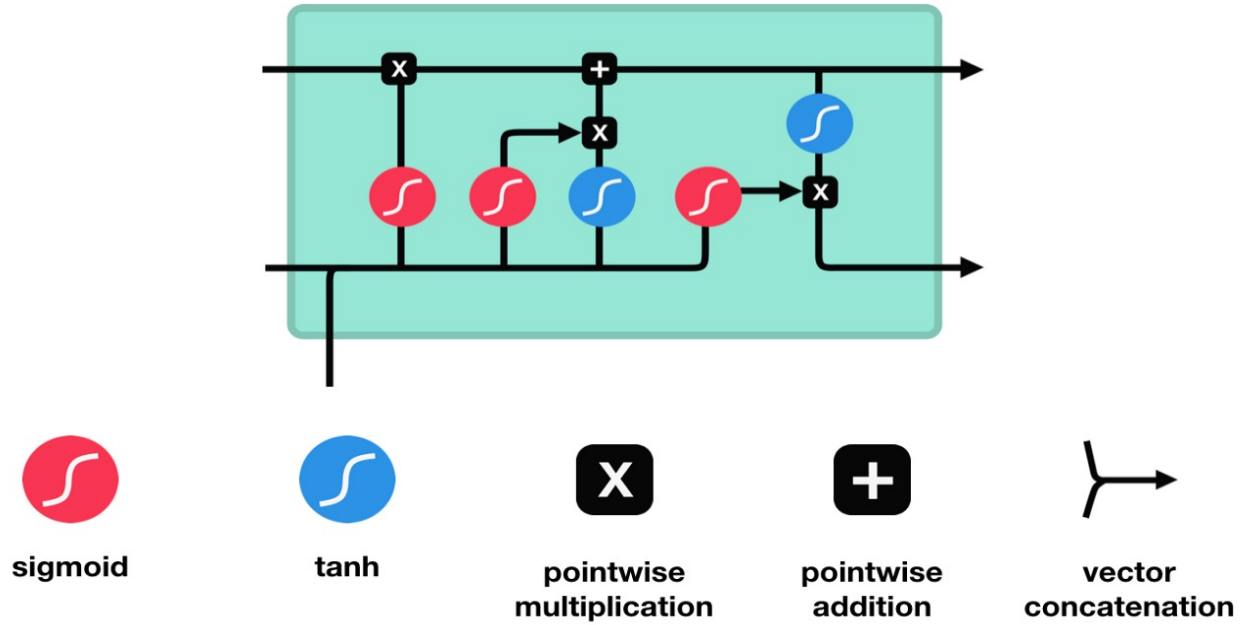
- An **LSTM cell** consists of a cell state and input, forget and output gates which make use of several activation functions.

- Both RNNs and LSTMs pass data as it propagates forward. However, unlike in RNNs, LSTMs makes use of gates to decide if it should keep or forget information.

LONG SHORT TERM MEMORY (LSTM)

STAATLICH
ANERKANNTE
HOCHSCHULE

LSTM contains computational blocks to control the information flow.



LSTM cell blocks are able to track information through many timesteps.

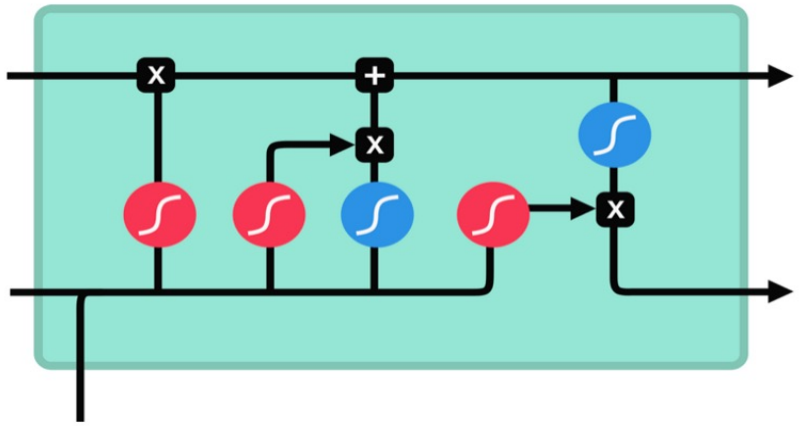
LONG SHORT TERM MEMORY (LSTM)

STAATLICH
ANERKANNTE
HOCHSCHULE

How LSTM works ?

1. Forget 2. Store 3. Update 4. Output

Information is added or removed by via a structure called as gates.

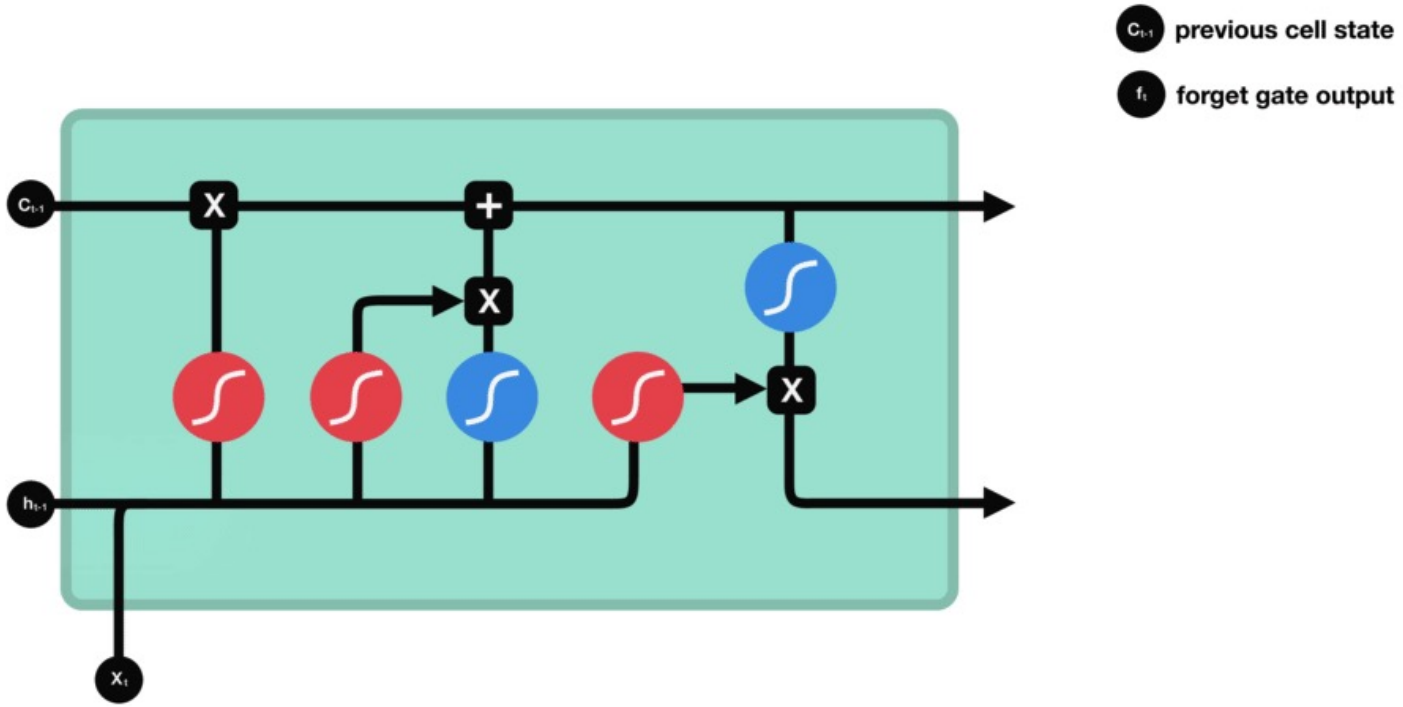


LSTM FORGET

STAATLICH ANERKANNTE HOCHSCHULE

LSTM forgets the irrelevant parts of previous steps.

- Consider, h(t−1) and x(t) concatenation, and outputs a number between 0 and 1 for each number in the cell state
- 1 = Completely keep this
- 0 = Completely forget this

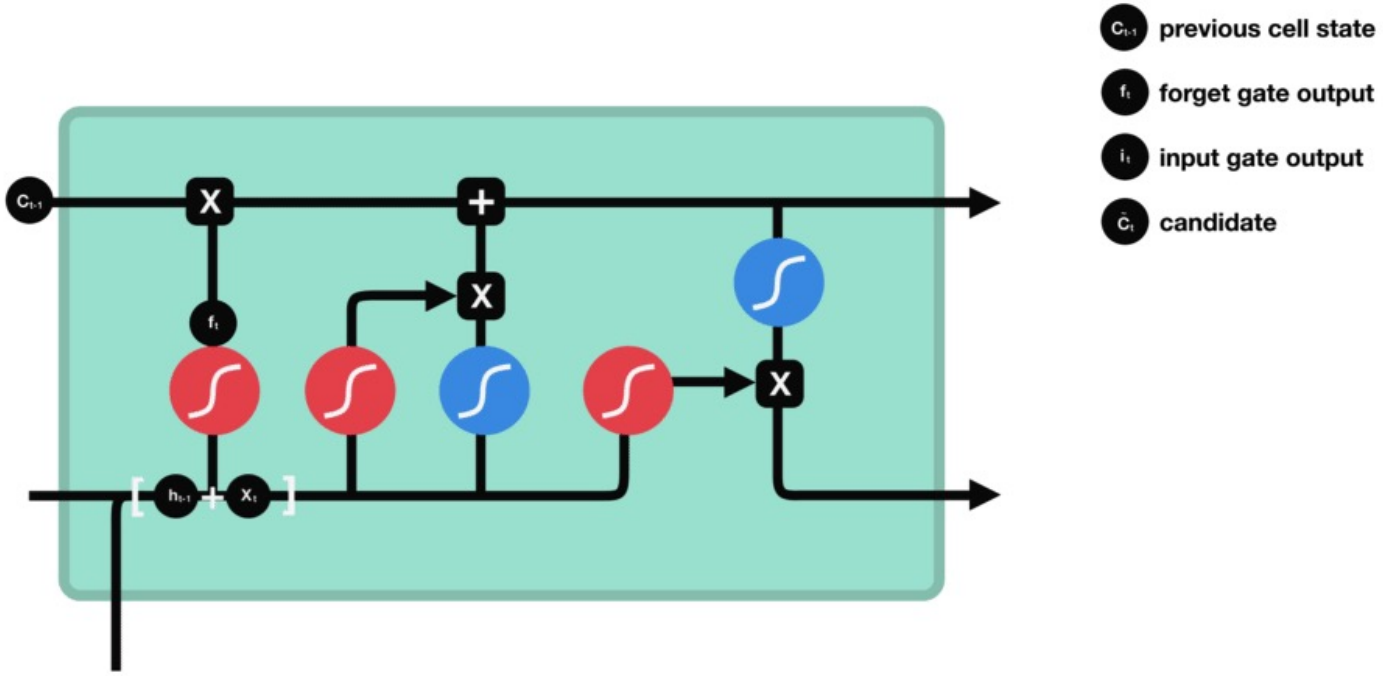


LSTM INPUT GATE

STAATLICH
ANERKANNTE
HOCHSCHULE

LSTM stores the relevant new information in the cell state.

- To update the cell state, we have the input gate which decides which values will be updated.
- A tanh layer creates a vector of new candidate values which will be carried forward.

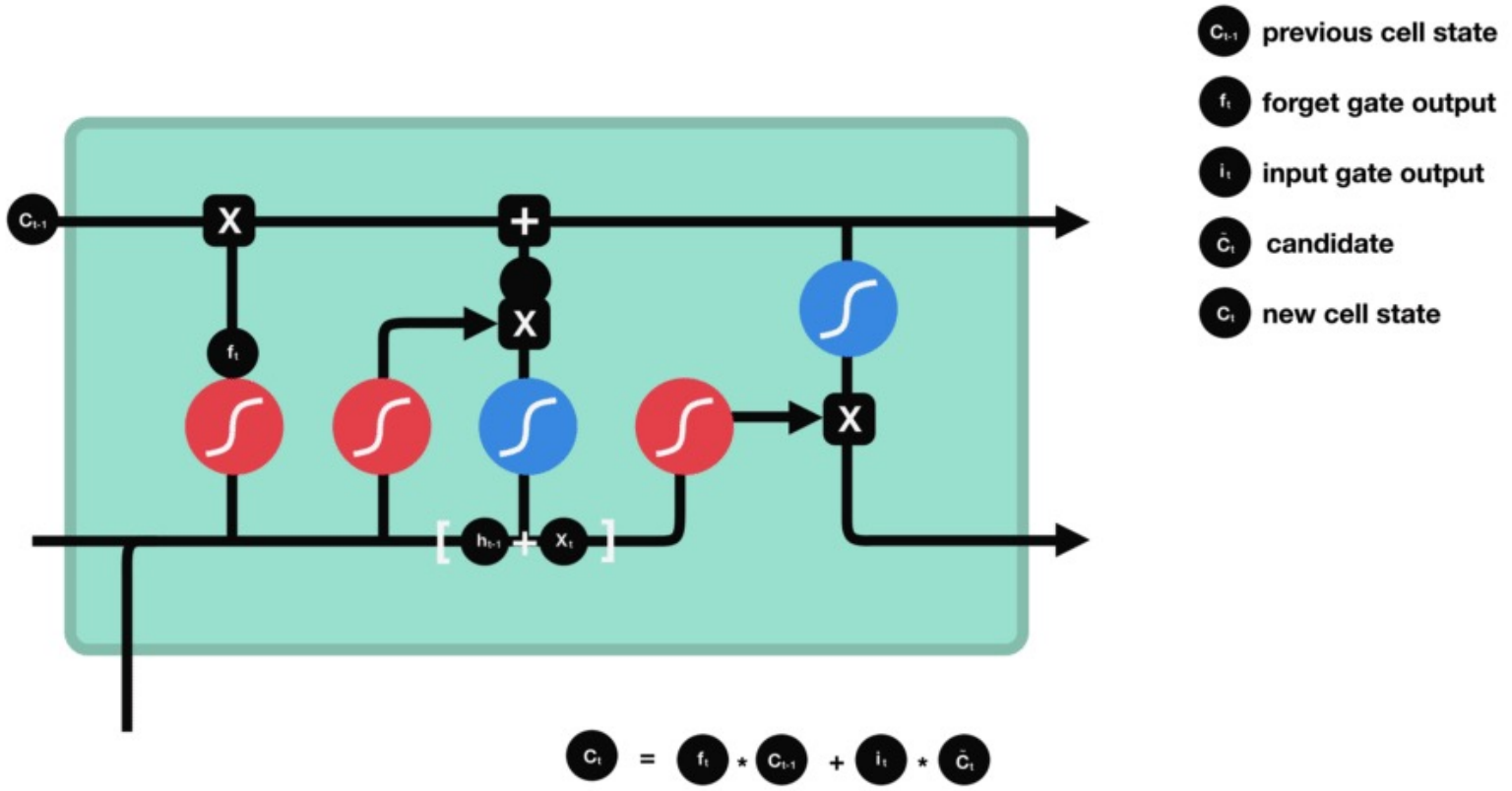


LSTM STORE/CELL STATE

STAATLICH ANERKANNTE HOCHSCHULE

LSTM selectively update cell state value

- Multiplication f_t and cell state of previous cell $C_{(t-1)}$is done.
- Then we add the current store state creating the current state C_t

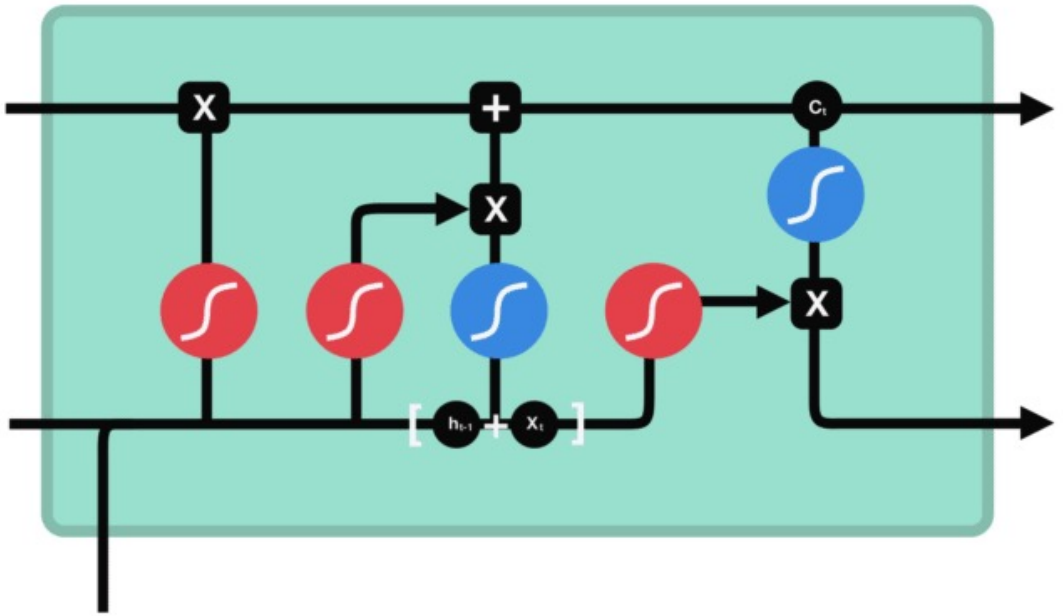


LSTM OUTPUT

STAATLICH
ANERKANNTE
HOCHSCHULE

The output gate controls what information is sent to the next step

- We run a sigmoid layer which decides what parts of the cell state we're going to output. The things we remember.

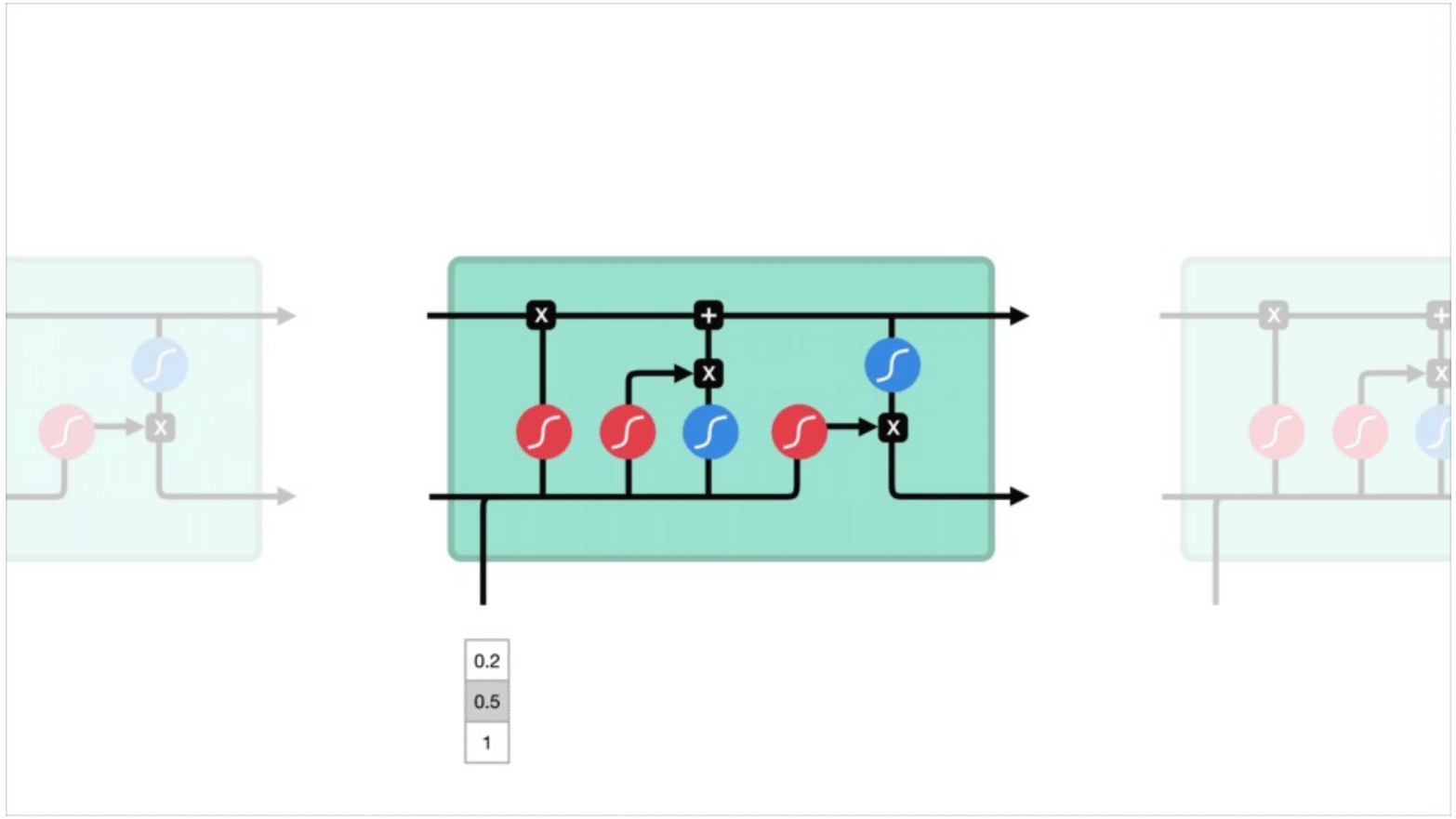


C_{t-1} previous cell state
f_t forget gate output
i_t input gate output
$\tilde{C}_t$ candidate
C_t new cell state
o_t output gate output
h_t hidden state

LONG SHORT TERM MEMORY

STAATLICH
ANERKANNTE
HOCHSCHULE

1. Forget 2. Store 3. Update 4. Output



LSTM CELL IN A FUNCTION

STAATLICH
ANERKANNTE
HOCHSCHULE

```
def LSTMCELL(prev_ct, prev_ht, input):
    combine = prev_ht + input
    ft = forget_layer(combine)
    candidate = candidate_layer(combine)
    it = input_layer(combine)
    Ct = prev_ct * ft + candidate * it
    ot = output_layer(combine)
    ht = ot * tanh(Ct)
    return ht, Ct


ct = [0, 0, 0]
ht = [0, 0, 0]
for input in inputs:
    ct, ht = LSTMCELL(ct, ht, input)
```

LSTM HYPERPARAMETERS TUNING TIPS

STAATLICH
ANERKANNTE
HOCHSCHULE

- Regularization helps: regularization methods include l1, l2, and dropout among others.
- The larger the network, the more powerful, but it’s also easier to overfit.
- More data is almost always better, because it helps fight overfitting.
- Train over multiple epochs (complete passes through the dataset).
- Evaluate test set performance at each epoch to know when to stop (early stopping).

RNN AND LSTM APPLICATIONS

STAATLICH
ANERKANNTE
HOCHSCHULE

Music Generation

Input – Sheet Music

Output – Next Character in sheet music

Classic Example of many to many with input length != Output length.

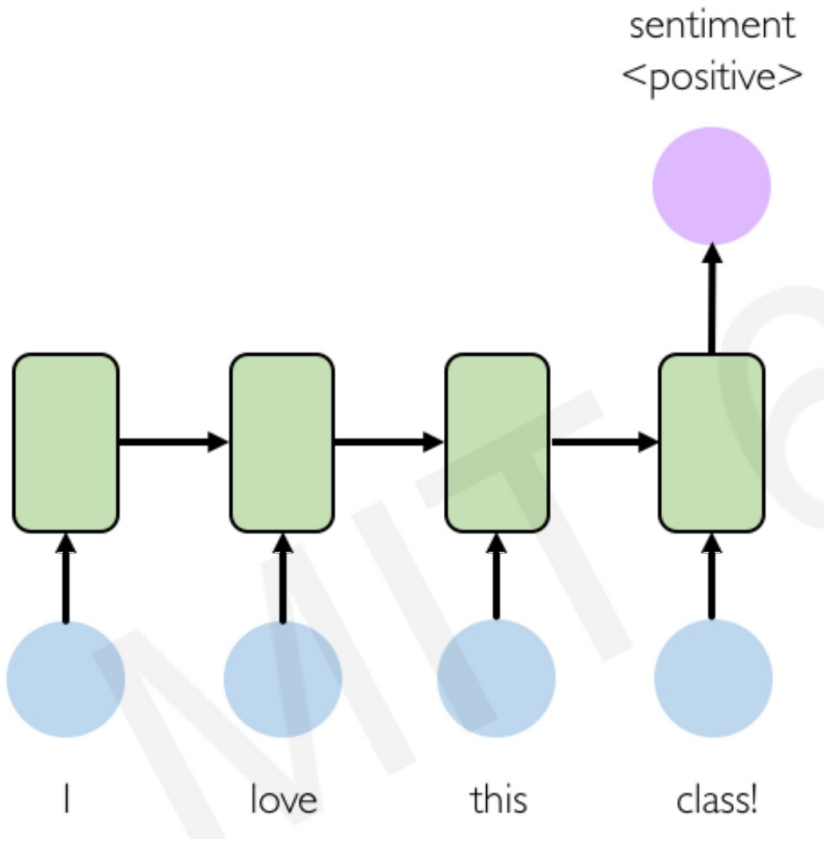
RNN AND LSTM APPLICATIONS

Sentiment Classification

Input – Sequence of words

Output - Probability of the sentiment.

Classic Example of many to 1.



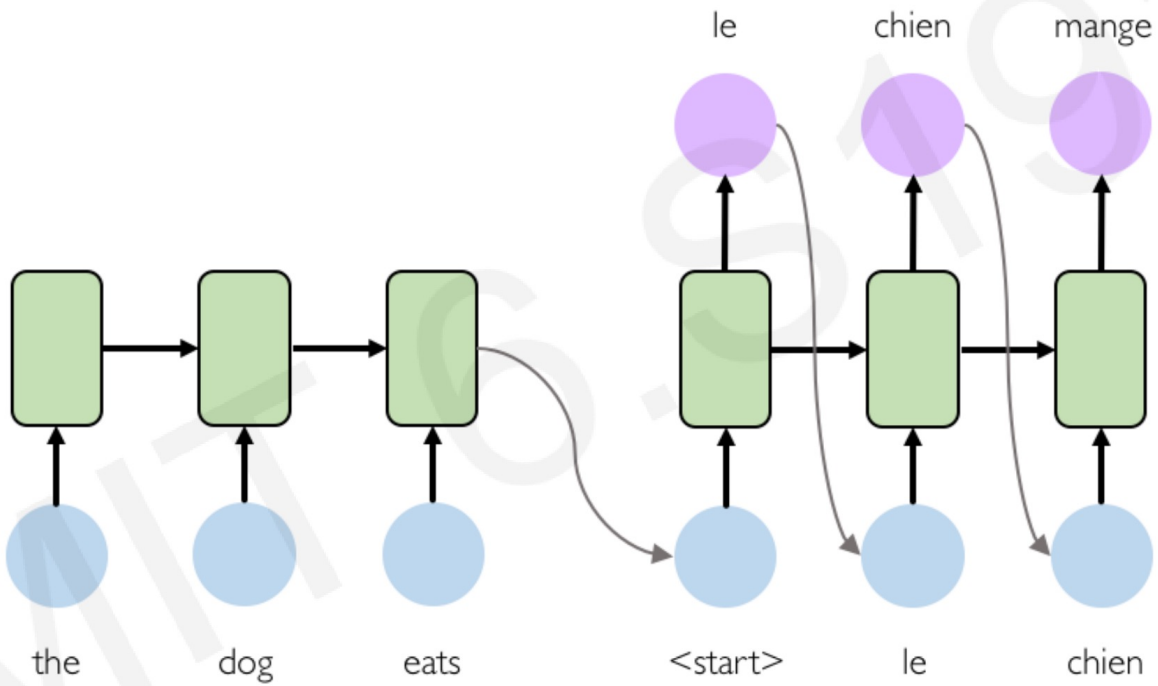
RNN AND LSTM APPLICATIONS

STAATLICH ANERKANNTE HOCHSCHULE

Machine Translation

Input – Sequence of words in origin language.

Output – Sequence of words in target language.



English (Encoder)

French (Decoder)

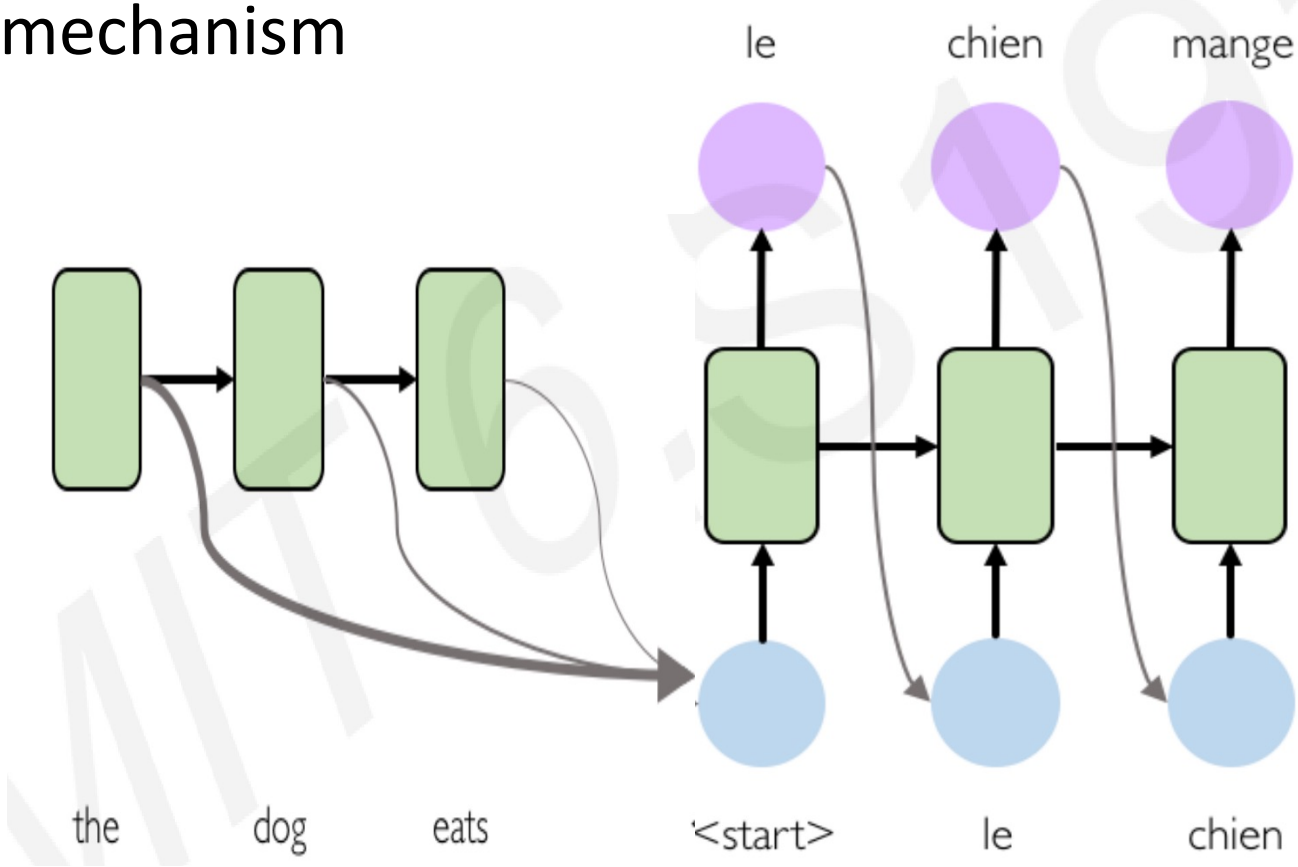
RNN AND LSTM APPLICATIONS

STAATLICH ANERKANNTE HOCHSCHULE

Machine Translation with attention mechanism

Input – Sequence of words in origin language.

Output – Sequence of words in target language.

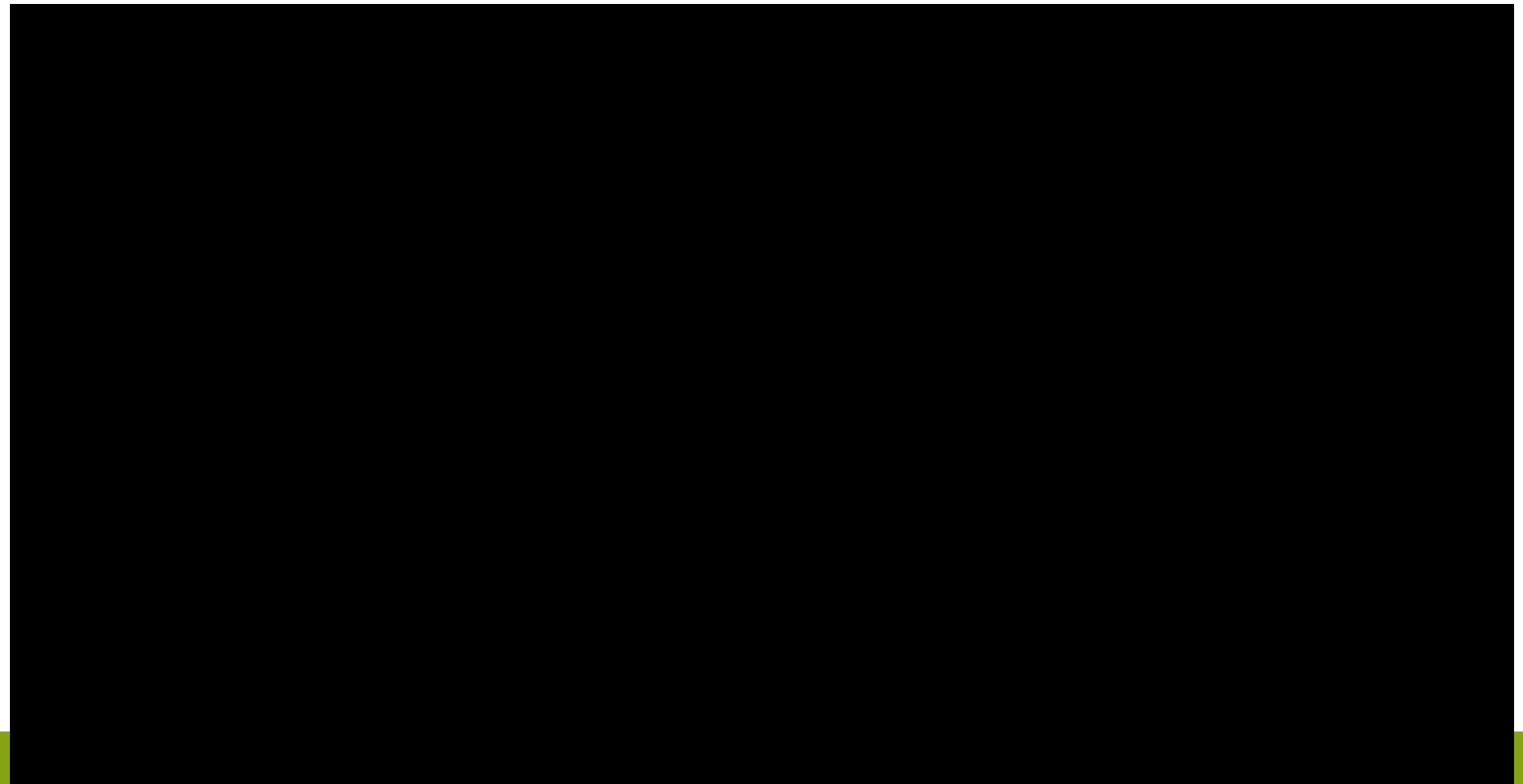


English (Encoder)

French (Decoder)

RNN AND LSTM APPLICATIONS

Trajectory Prediction – Self-Driving Cars



RNN AND LSTM APPLICATIONS

STAATLICH
ANERKANNTE
HOCHSCHULE

Environmental Modeling

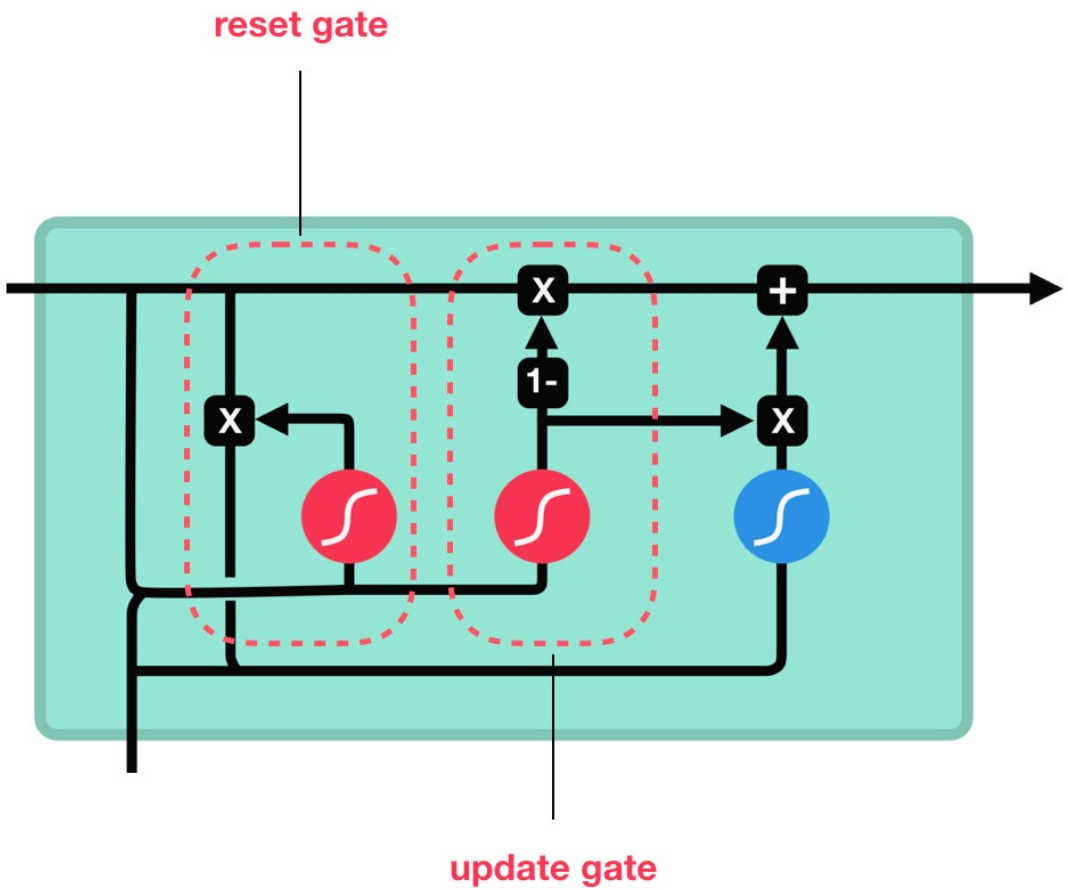
GATED RECURRENT UNITS (GRU)

STAATLICH ANERKANNTE HOCHSCHULE

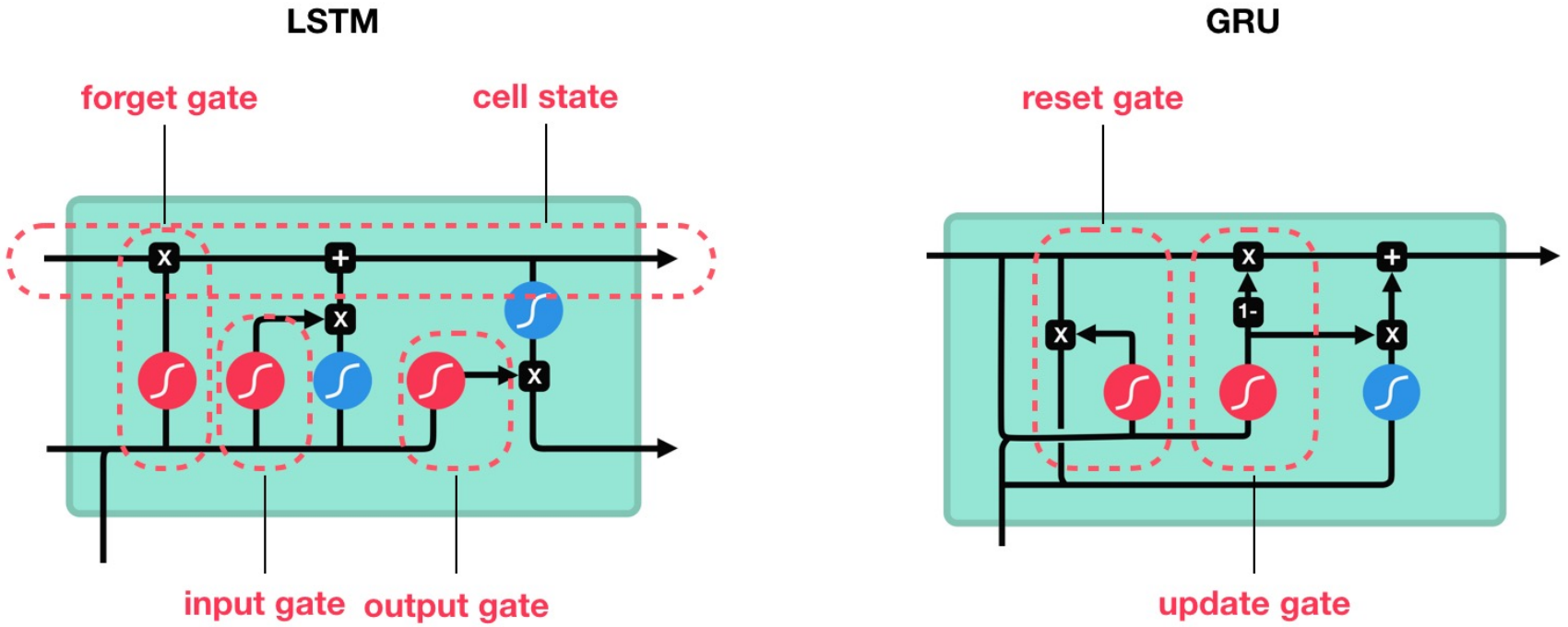
1. **Gated recurrent unit**s (**GRU**s) are a gating mechanism in recurrent neural networks, introduced in 2014 by Kyunghyun Cho. [paper]

2. The GRU is like a long short-term memory (LSTM) with forget gate but has fewer parameters than LSTM

3. The GRU does not have any output gate as opposed to the LSTM.



LSTM VS GRU

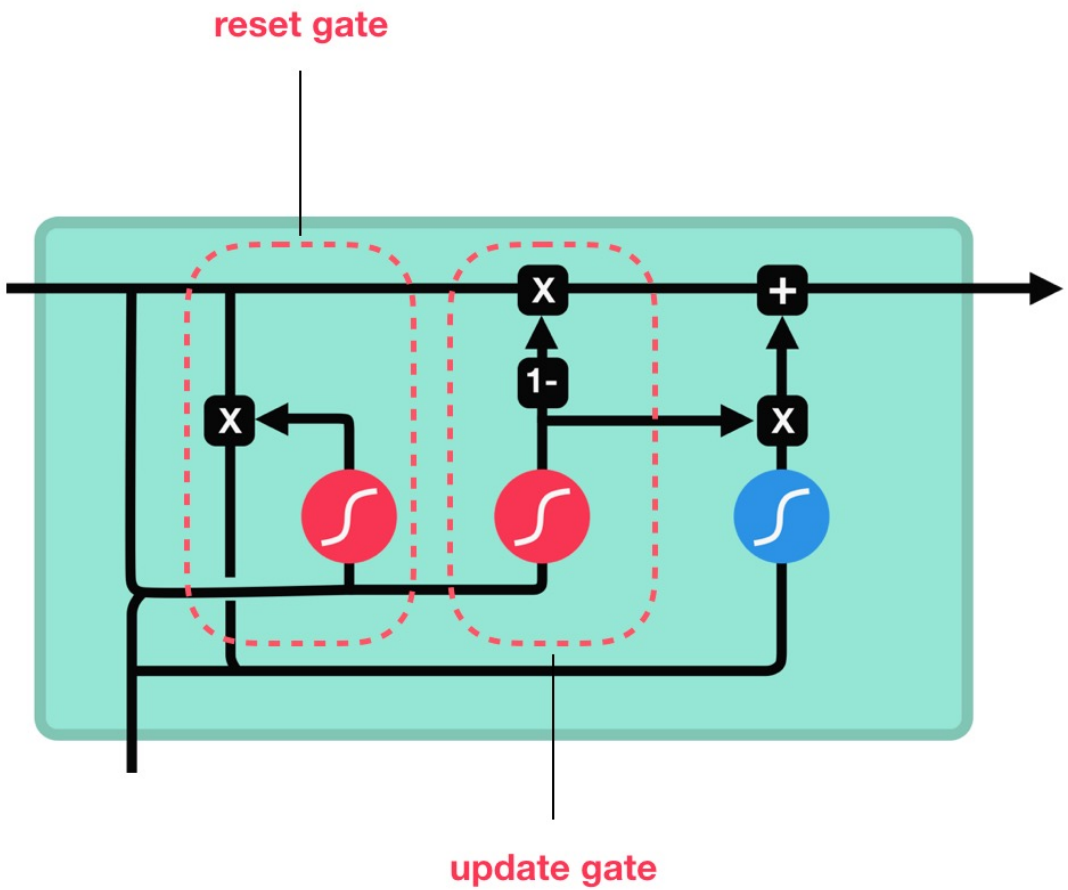

LSTM
forget gate
cell state
input gate
output gate
GRU
reset gate
update gate
1-


sigmoid
tanh
pointwise multiplication
pointwise addition
vector concatenation

GRU ARCHITECHTURE

STAATLICH
ANERKANNTE
HOCHSCHULE

1. GRU is similar in structure as compared to LSTM
2. GRU got rid of cell state and used hidden state to transfer information
3. Replaced the forget and input gate with update gate
4. reset gate is another gate is used to decide how much past information to forget



GRU SUMMARY

- Gated recurrent neural networks are better at capturing dependencies for time series with large timestep distances.

- Reset gates help capture short-term dependencies in time series.

- Update gates help capture long-term dependencies in time series.

GRU LIMITATIONS

• The LSTM is "strictly stronger" than the GRU the GRU fails to learn simple languages that are learnable by the LSTM as proved by Chung, Junyoung; Gulcehre, Caglar in their paper “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”.

• Due to fewer gates the computation is faster but the control over learning mechanism is also traded-off.

• Similarly, as shown by Denny Britz & Anna Goldie & Minh-Thang Luong & Quoc Le of Google Brain, LSTM cells consistently outperform GRU cells in "the first large-scale analysis of architecture variations for Neural Machine Translation "

HOME WORK ASSIGNMENTS – 1

STAATLICH
ANERKANNTE
HOCHSCHULE

- Sentiment analysis of IMDB data.

  - By performing NLP based analysis predict the sentiment of the movie review use RNN and LSTM to compare.

  - Dataset available at - https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

  - Do not copy from other kernels (They are visible to US as well ☺).

HOME WORK ASSIGNMENTS – 2

STAATLICH
ANERKANNTE
HOCHSCHULE

- "How much did it rain?" dataset on Kaggle (https://www.kaggle.com/c/how-much-did-it-rain-ii)

- The task is a time series prediction. You have snapshots of polarimetric radar values and you have to predict the hourly rain gauge total. (some insights from the runner-up in the competition)

- Do not copy from other kernels (They are visible to US as well ☺).

THANK YOU.

STAATLICH
ANERKANNTE
HOCHSCHULE