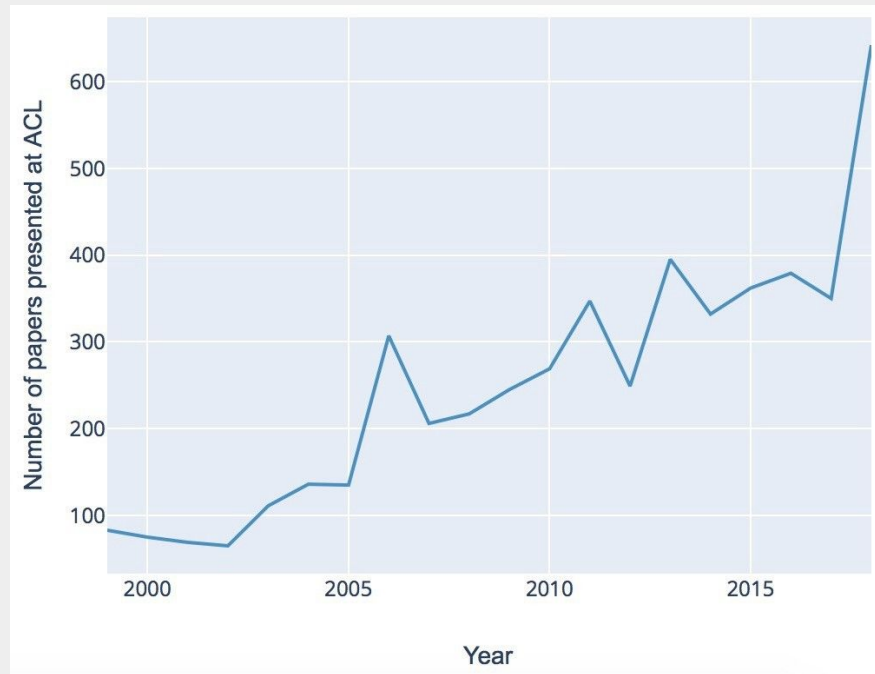


Recurrent Neural Network based Text Generation

Koen Vannisselroij and Sonali Fotedar

Generating Text

- Gaining popularity
- Text Summarization
- Writing Annual Reports
- Twitterbots
- Conversational Marketing



Annual Meeting of the Association
for Computational Linguistics (ACL)

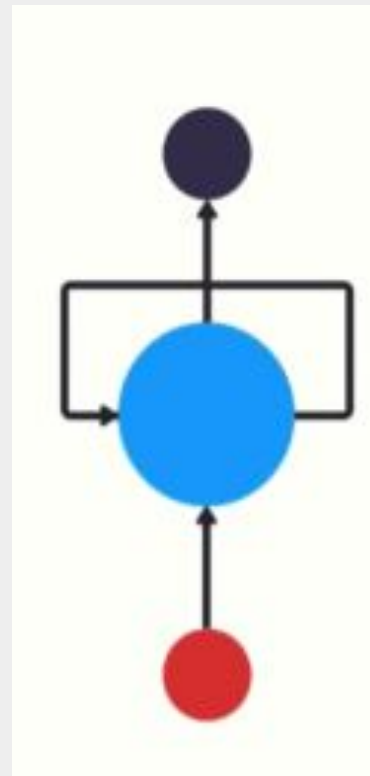
Understanding Sequential Data

- Any data that is a **sequence of data points** of which the previous data points are **relevant for future** datapoint prediction (text, weather, stock price charts, audio)
- Computer readable format = numbers

wiped by an almost perfectly symmetrical
leano. The ceaseless beat of minute waves
the rocks I sit on, make me increasing
every and ^{relaxed} I saw my dreams today...
ling hills, covered with small yellow
downs, cut off abruptly, as cliffs in
the sea. I am in awe. Looking off
to the distance... snow capped mo
to the sky, beautifully rolling but smoothly
orned clouds. Vaguely appearing like
ondore in ~~the~~ flight. The sun
is still out, and is beginning to set. The
now on the Orono Volcano turns orange, and
see the sky on the opposite horizon. As
walk back through the Plaza, I see the
his must be the hang-out area for the
young people at night. At the same time it
is where the older generation takes its walks.
Interesting and unusual combination. That

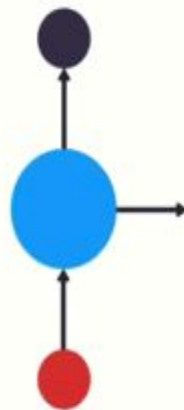
Recurrent Neural Networks

- First input: the input is a word which gets fed in. Then an output value is calculated based on this word.
- For the second word, the information of the previous word is also given as an input. The previous output is recurrent. Then based on both these inputs, an output is given. Which in turn serves as an input to the next state.



Recurrent Neural Networks

- If we unroll the recurrent block it would look like this. Where the red dots are the input and the black dots the output.
- The output is fed to a function called softmax, which assigns a probability to every word that is present in our vocabulary (the datafile used to train the network).

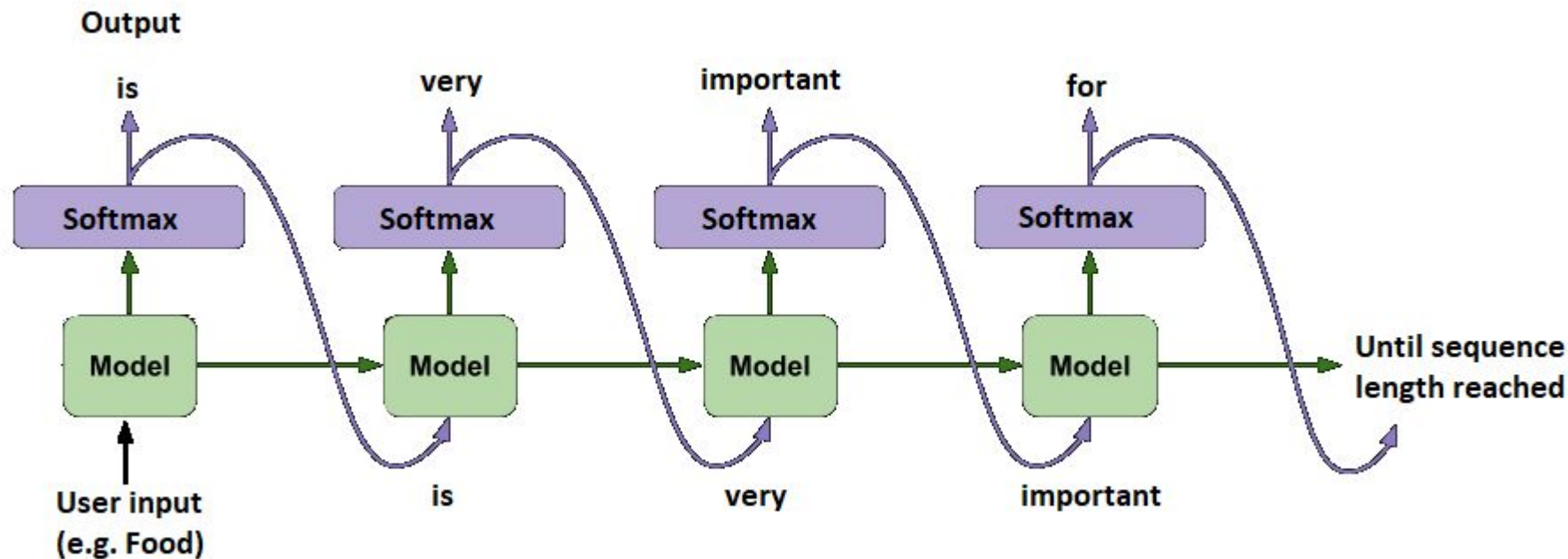


Categorical cross-entropy

- The vector of probabilities resulting from this function is evaluated by a so-called loss function, in our case we use categorical cross-entropy to evaluate how well the output was. It checks the ground-truth value and compares this to the softmax output.
- Using backpropagation the weights of the network are optimized.

$$CE = - \sum_i^C t_i \log(f(s)_i)$$

Generating Sequences of Text



**CAN'T FIND THE WORDS
HE'S LOOKING FOR**

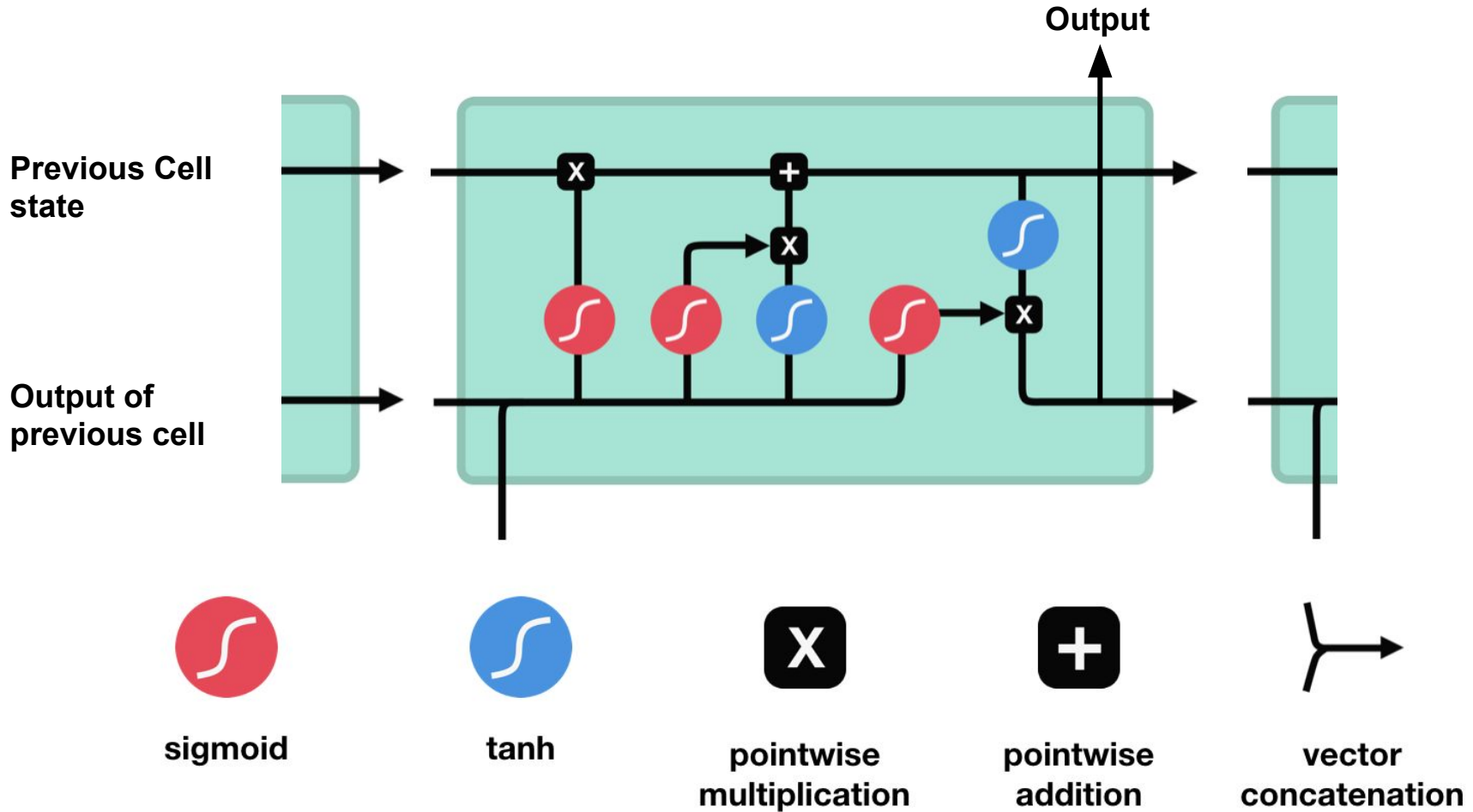
GENERATES THEM

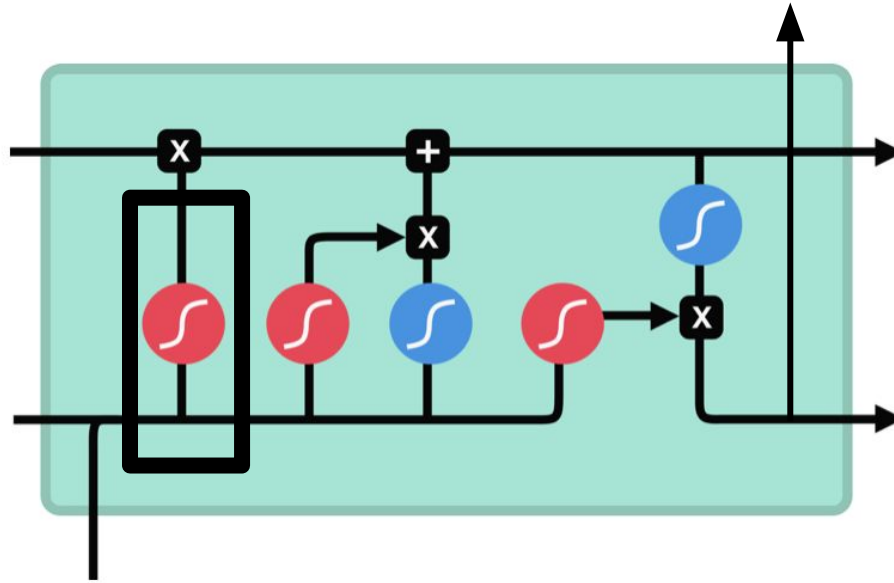
WORKSHOP

Problems with RNN

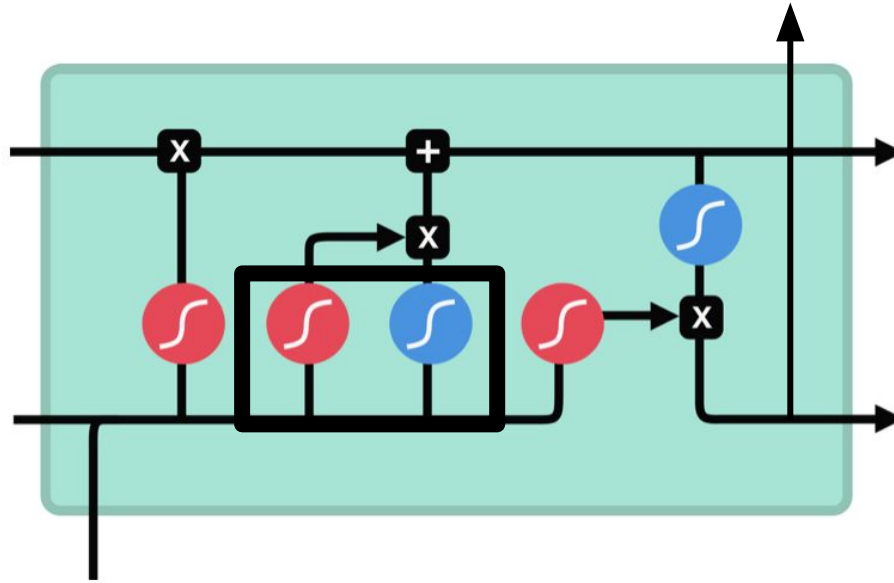
Long-term dependencies

[Andreas Madsen](#)

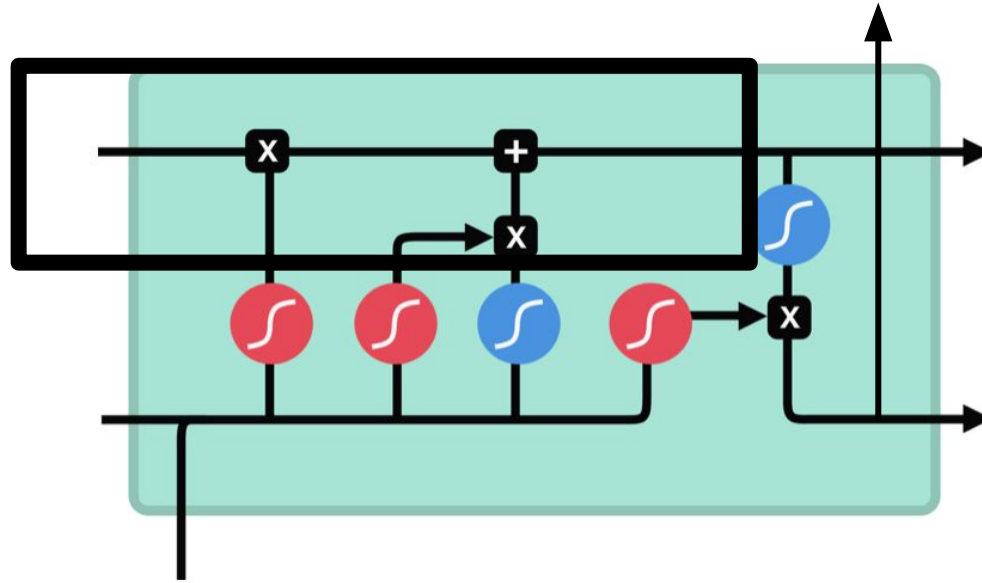




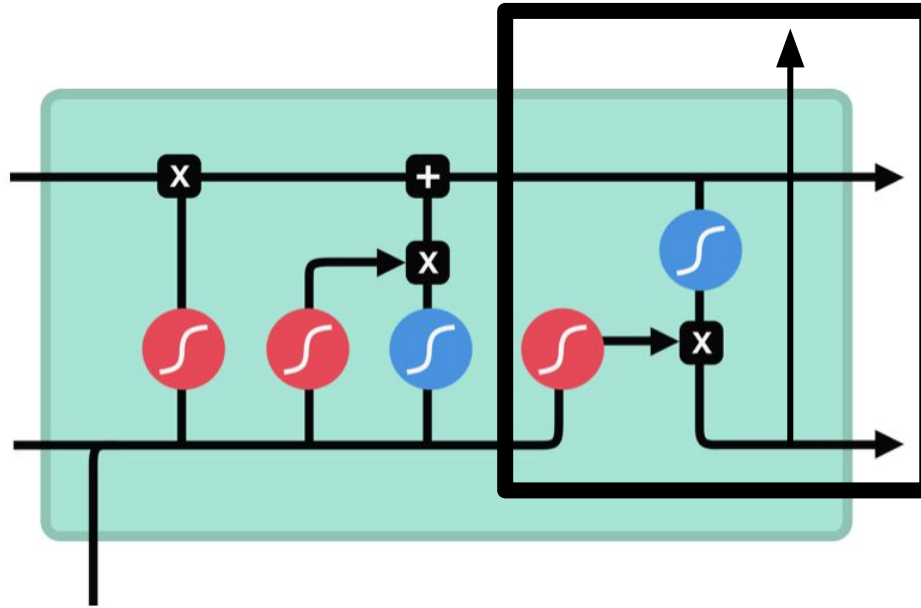
The forget gate determines if the information that the LSTM-block remembered for the last input is still relevant for the new input. It takes the previous output in account when determining what information is worth keeping.



The update gate determines which part of the new information should be added to the cell state. After selecting which values to update it creates values for the input that is worth remembering.



The pipeline for the cell state runs the previous cell state through the values calculated by the forget gate, removing information that is not worth to remember, and then adds the information that is worth remembering for the next input. After this the cell state is given to the next cell as input.



The output gate uses the input of the previous block to determine what needs to be given as an output, taking into account what the cell state has remembered when creating an output. Then the output goes to the next layer of the network, and serves as input to the next state of the next LSTM cell.

Further Reading

[Understanding LSTM Networks](#)

[General Text Generation](#)

[Attention](#)

[Transformer Models](#)