

IFI 9000 Analytics Methods Neural Networks & Deep Learning

by **Houping Xiao**

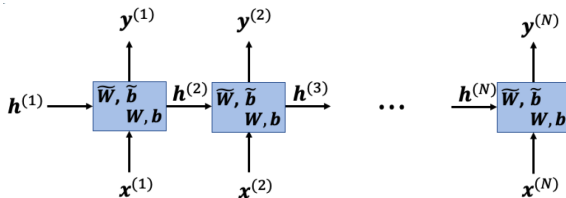
Spring 2021



Introduction

Recurrent neural networks

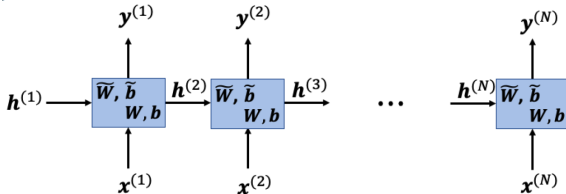
- While CNNs work quite promising for images, they may not be the best modeling tools for other data sets such as time series data
- For temporal, or time-series data and stream inputs (e.g., text streams), recurrent neural networks (RNNs) are of major attention



- We assume a sequence of data is streamed as N time instances, and mapped to a sequence of response (here of the same length).
- For now let's assume that the input and output have similar lengths

RNN: governing equations

- Remember in standard neural network the output of the hidden layer was in the form $\mathbf{h} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$



- In RNNs the input is a stream $\mathbf{x}(t)$ and we have another coefficient matrix that makes the current hidden output dependent on the previous one:

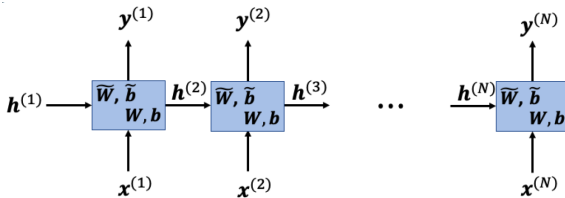
$$\mathbf{h}^{(t)} = \sigma \left(\mathbf{W} \begin{pmatrix} \mathbf{h}^{(t-1)} \\ \mathbf{x}^{(t-1)} \end{pmatrix} + \mathbf{b} \right),$$

$$\mathbf{y}^{(t)} = \sigma \left(\tilde{\mathbf{W}}\mathbf{h}^{(t)} + \tilde{\mathbf{b}} \right), t = 1, \dots, N$$

- Training cost per sample: $\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{t=1}^N L(\mathbf{y}^{(t)}, \hat{\mathbf{y}}^{(t)})$

Types of RNN and applications

- The following architecture is **many-to-many**, with the input and output having the same length

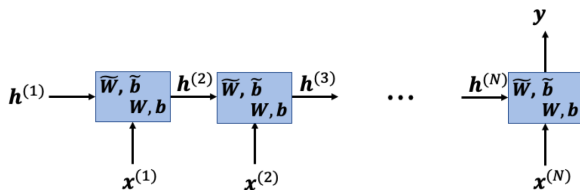


- Application example is named-entity recognition (classify unstructured text into predefined classes)

contentSkip to site indexPublicsSubscribeLog InSubscribeLog InToday's PaperAdvertisementSupported ORG byf B.I. Agent Peter Sztrok PERSON .
Who Colicized Trump PERSON in Texts, to FredImagePeter Sztrok, a top F.B.I. OPE counterintelligence agent who was taken off the special counsel investigation after his disparaging texts about President Trump PERSON were uncovered, was fired. CreditJ. Kingpatrick PERSON for The New York TimesBy Adam Goldman ORG and Michael S. SchmidtAug PERSON 13 CARDINAL . 2018WASHINGTON CARDINAL — Peter Sztrok PERSON the F.B.I. OPE senior counterintelligence agent who disparaged President Trump PERSON as inflammatory text messages and helped oversee the Hilary Clinton PERSON email and Russia OPE investigations, has been fired for violating bureau policies, Mr. Sztrok PERSON 's lawyer said Monday DATE Mr. Trump and his allies seized on the texts — exchanged during the 2016 DATE campaign with a former F.B.I. OPE lawyer, Lisa Page in PERSON assailing the Russia OPE investigation as an illegitimate "witch hunt." Mr. Sztrok PERSON , who rose over 20 years DATE at the F.B.I. OPE to become one of its most experienced counterintelligence agents, was a key figure in the early months DATE of the inquiry.Along with writing the texts, Mr. Sztrok PERSON was accused of sending a highly sensitive search warrant to his personal email account.The F.B.I. OPE had been under immense political pressure by Mr. Trump PERSON to dismiss Mr. Sztrok PERSON , who was removed last summer DATE from the staff of the special counsel, Robert S. Mueller III PERSON . The president has repeatedly denounced Mr. Sztrok PERSON in posts on

Types of RNN and applications

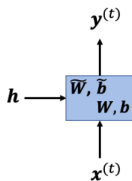
- The following architecture is **many-to-one**



- Application example is sentiment classification (review systems, scoring systems)

Types of RNN and applications

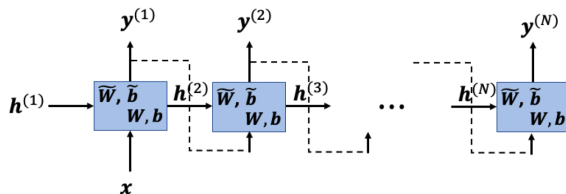
- The following architecture is **one-to-one**



- This is somehow equivalent to traditional one-layer network (real-time mapping)

Types of RNN and applications

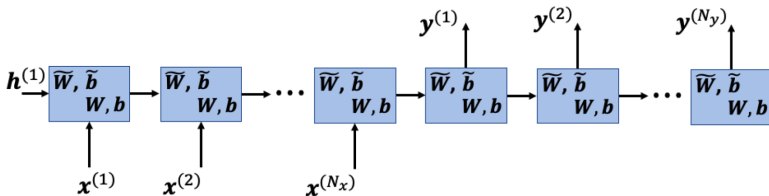
- The following architecture is **one-to-many**



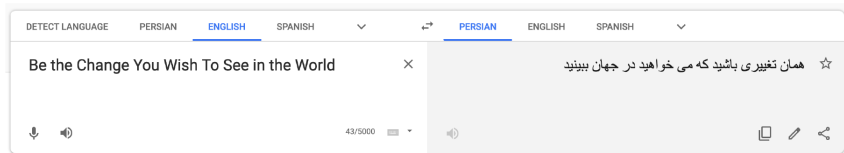
- Application example is music generation, image captioning

Types of RNN and applications

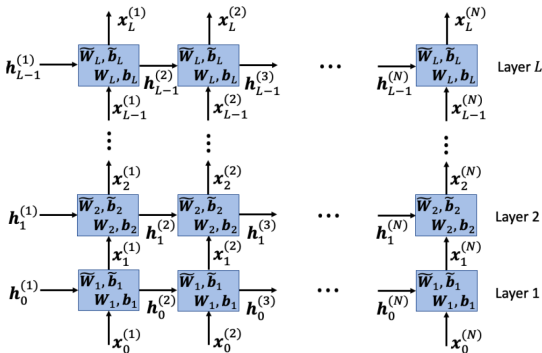
- The following architecture is **many-to-many**, with the input and output having different lengths



- Application example is machine translation, video captioning



- All the architectures we explained so far can become deep and layered



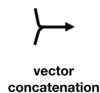
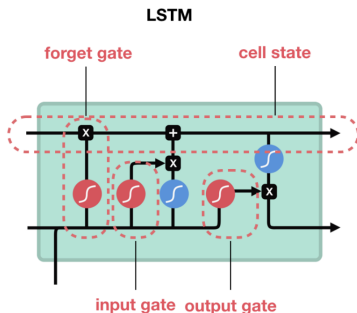
- In practice we do not need very deep RNNs (unlike standard DNNs which can be very deep)

Problems with standard RNNs and remedies

- Hard to train and vanishing gradient
- Difficulty accessing information from long time ago
- Two main variants of RNNs:
 - Long Short-Term Memory (LSTM)
 - Gated Recurrent Units (GRUs)
- To learn more and see some cool applications see:
<https://www.youtube.com/watch?v=6niqTuYFZLQt=1850s>

LSTM: Long Short-term Memory

- LSTM developed for resolving short-term memory, i.e., RNN can forget what it seen in longer sequences
- LSTM uses “gates” to regulate the flow of information, keep important information or throw away unimportant information



LSTM: a toy example

Customers Review 2,491

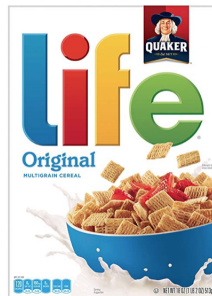


Thanos

September 2018

Verified Purchase

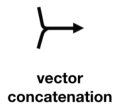
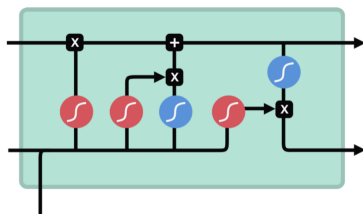
Amazing! This box of cereal gave me a perfectly balanced breakfast, as all things should be. I only ate half of it but will definitely be buying again!



A Box of Cereal
\$3.99

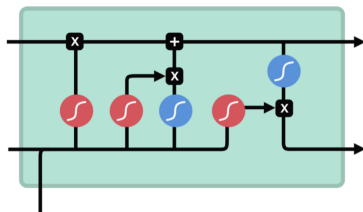
LSTM

- Keep or forget information using gates
- Gates are different neural networks, contains sigmoid functions
- Sigmoid squishes values between 0 (forgotten) and 1 (kept).



LSTM: forget gate

- Based on the previous hidden state and current input, the forget gate decides what information should be forgotten or kept
- output value closer to 0 means to forget, and the closer to 1 means keep



sigmoid



tanh



pointwise
multiplication



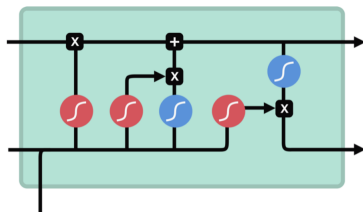
pointwise
addition



vector
concatenation

LSTM: input gate

- Based on the previous hidden state and current input, the input gate uses sigmoid to decide what information should be updated
- uses tanh function to regulate the network
- the sigmoid output will decide which information is important to keep from the tanh output



sigmoid



tanh



pointwise
multiplication



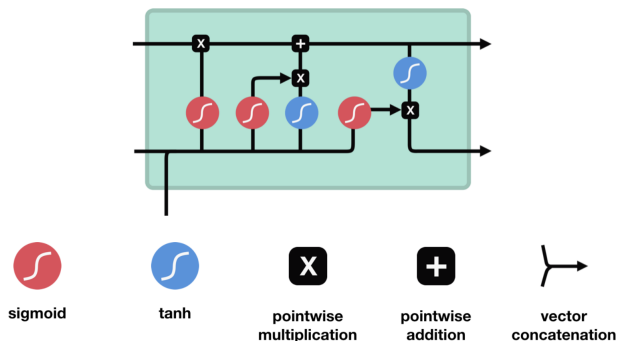
pointwise
addition



vector
concatenation

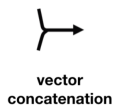
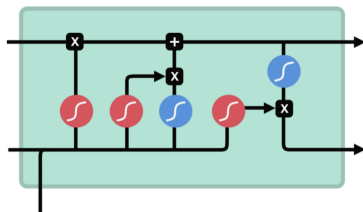
LSTM: cell state

- Based on the previous cell state, output from the input gate, to update the new cell state.



LSTM: output gate

- Output the new cell state
- output the hidden state, based on the previous hidden state and the current input



The End