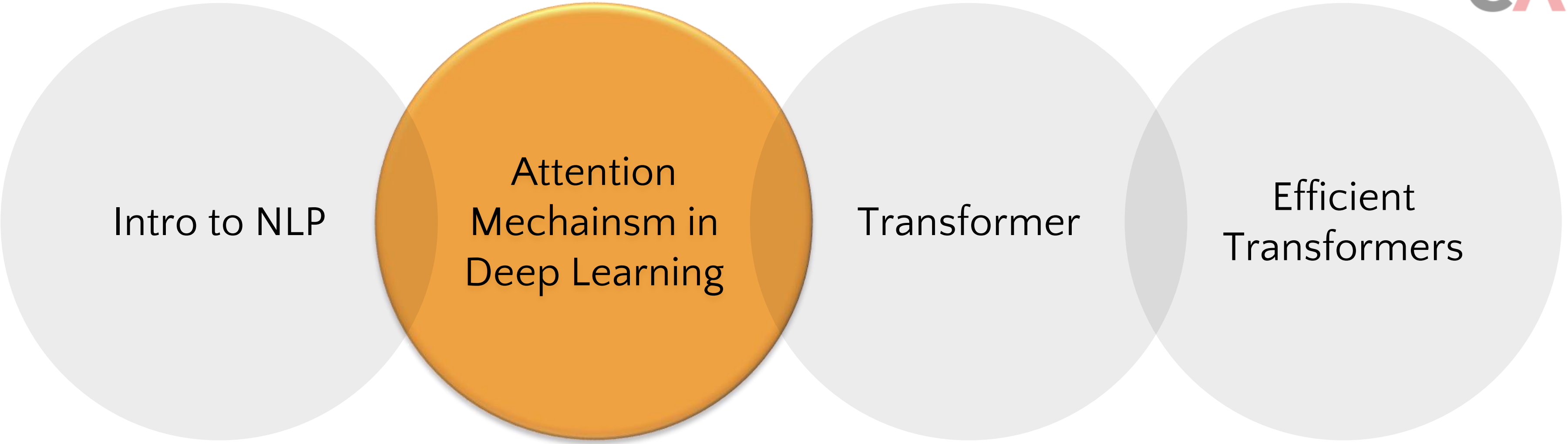


Intro to NLP

Attention Mechainsm in Deep Learning

Transformer

Efficient Transformers

GitHub

colab

Contents

Proposed by ChatGPT

1. Introduction to deep learning and its applications
2. Overview of recurrent neural networks and its limitations
3. Introduction to attention mechanism and its benefits
4. Explanation of the mechanism behind attention with mathematical formulation
5. Demonstration of attention mechanism in code
6. Explanation of self-attention and its application in transformers
7. Comparison of RNNs and transformers in terms of performance and efficiency
8. Demonstration of transformer in a real-world application
9. Conclusion and future directions in attention mechanism research.

Why Attention & Transformers

Why should I learn how transformers and attention mechanism work?

Here is why according to ChatGPT

“Here are a few reasons why learning about transformers and attention mechanisms could be valuable:

1. **Advancements:** Transformers and attention mechanisms are the **state-of-the-art** techniques in deep learning, and knowledge in these areas can help you keep up with advancements and developments in the field.
2. **Real-world applications:** Transformers and attention mechanisms have a **wide range of applications, from NLP and speech recognition to computer vision and robotics.** Understanding these techniques can help you build innovative solutions to real-world problems.
3. **Career advancement: Knowledge of transformers and attention mechanisms is highly valued in many industries**, and having this skill set could help you advance your career and increase your earning potential.
4. **Problem-solving:** Understanding these techniques can **help you think about and approach problems from a new perspective,** and give you the tools to build more effective solutions.
5. **Better intuition:** Learning about transformers and attention mechanisms can **help you develop a deeper intuition for deep learning and machine learning** more broadly.”

What is a Transformer?

or

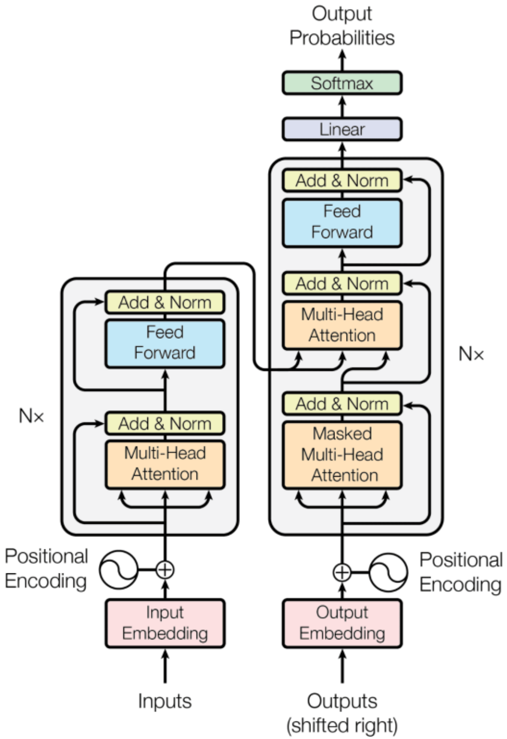
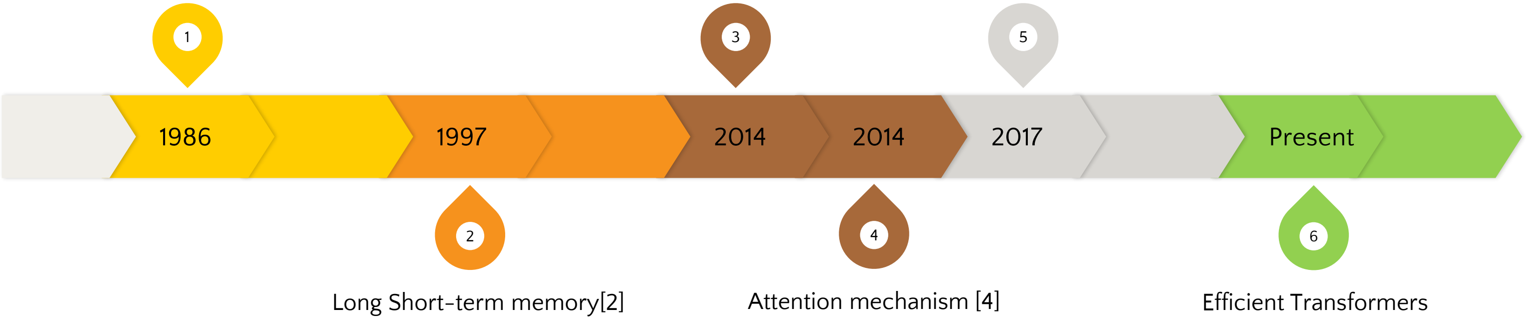


Image: https://pixabay.com/users/vinsky2002-1151065/

Text-to-image

DALL.E 2

History of Sequential Models



Recurrent NN [1]

1

1986

2

1997

Long Short-term memory[2]

Gated Recurrent NN [3]

3

2014

4

2014

Attention mechanism [4]

Transformer [5]

5

2017

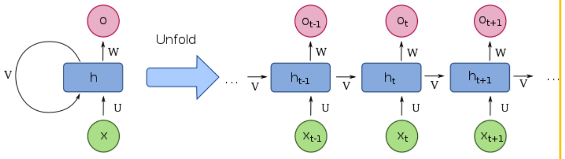
6

Present

Efficient Transformers

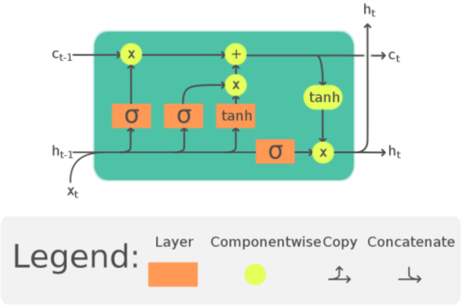
[1] Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. "**Learning representations by back-propagating errors**." *nature* 323, no. 6088 (1986): 533-536.
[2] Hochreiter, Sepp, and Jürgen Schmidhuber. "**Long short-term memory.**" *Neural computation* 9, no. 8 (1997): 1735-1780.
[3] Chung, Junyoung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. "**Empirical evaluation of gated recurrent neural networks on sequence modeling.**" *arXiv preprint arXiv*:1412.3555 (2014).
[4] Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "**Neural machine translation by jointly learning to align and translate**." *arXiv preprint arXiv*:1409.0473 (2014).
[5] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "**Attention is all you need**." *Advances in neural information processing systems* 30 (2017).

RNN, LSTM, GRU



RNN

the input at the time stamp and hidden state from the previous time step is passed through the activation layer to obtain a new state.

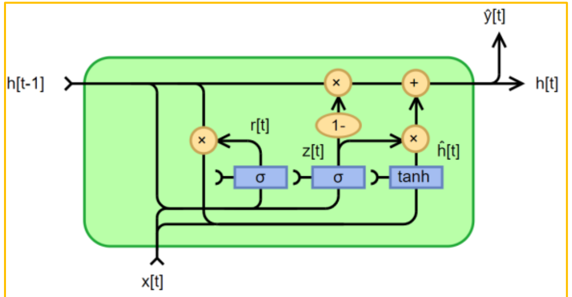


LSTM

Input gate: Decides what information will be stored in long term memory.

Forget Gate: Decides which information from long term memory be kept or discarded

Output gate: take the current input, the previous short-term memory and newly computed long-term memory to produce new short-term memory which will be passed on to the cell in the next time step.



GRU

Update gate: Determines the amount of previous information that needs to pass along the next state

Reset gate: Determines how much of the past information is needed to neglect

[1] https://en.wikipedia.org/wiki/Recurrent_neural_network
[2] https://en.wikipedia.org/wiki/Long_short-term_memory
[3] https://en.wikipedia.org/wiki/Gated_recurrent_unit

RNN vs LSTM vs GRU

RNN:

- Faces short-term memory problem due to vanishing gradient problem
- Slow training

LSTM:

- Fixes the vanishing gradient problem
- Requires more training data
- Slower

GRU:

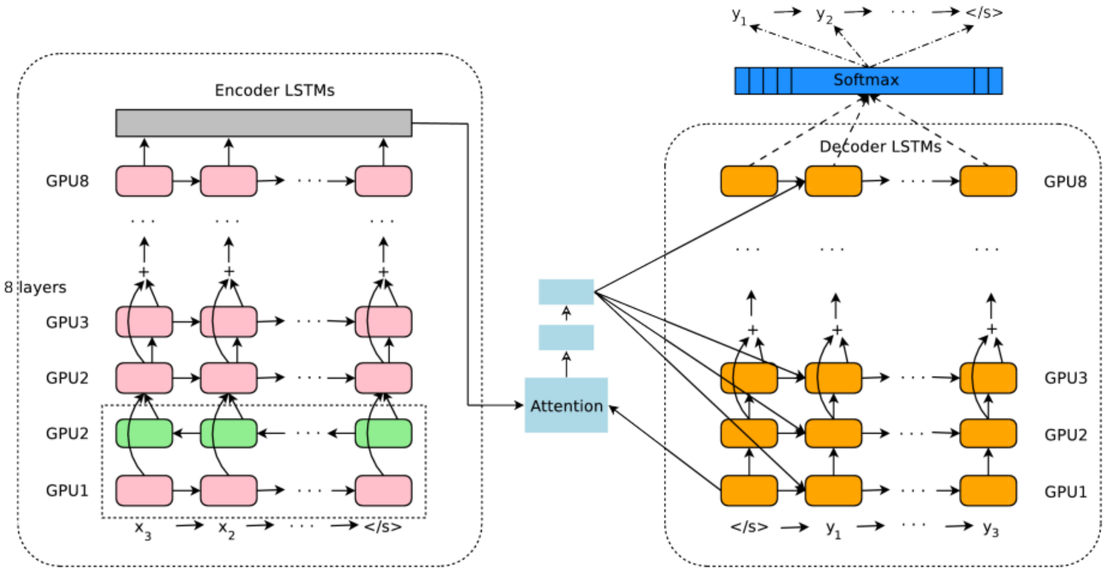
- Fixes the vanishing gradient problem
- Faster than LSTM

What is NN? https://www.youtube.com/watch?v=aircAruvnKk
https://jalammar.github.io/**visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention**/
Lipton, Zachary C., John Berkowitz, and Charles Elkan. "**A critical review of recurrent neural networks for sequence learning**." *arXiv preprint arXiv*:1506.00019 (2015).
Yu, Yong, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. "**A review of recurrent neural networks: LSTM cells and network architectures**." *Neural computation* 31, no. 7 (2019): 1235-1270.

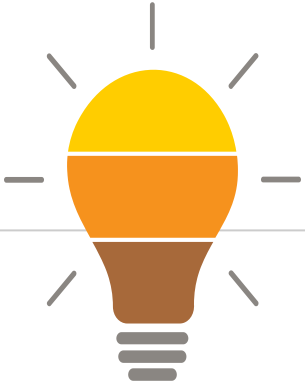
Visualizing seq2seq models

https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/

Ex: LSTM with Attention



Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun et al. "**Google's neural machine translation system: Bridging the gap between human and machine translation**." *arXiv preprint arXiv*:1609.08144 (2016).

Attention

Attention layers are fundamentally **a weighted mean reduction**. It is just computing a mean, where you somehow weight each element contributing to the mean. Since it is a mean, attention decreases the rank of an input tensor. The reduction occurs over the values; so, if the values are rank 3, the output will be rank 2. The query should be one less rank than the keys. The keys should be the same rank as the values. The keys and query determine how to weight the values according the **attention mechanism** -- a fancy word for equation.

https://github.com/whitead/dmol-book/blob/main/dl/attention.ipynb

Terminology

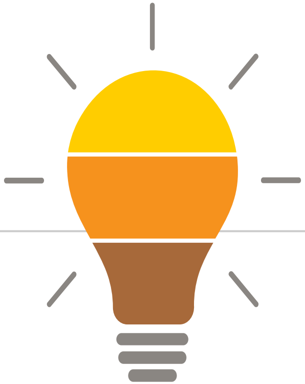
Key: Features of a certain area of the image, word embeddings of a document, or the hidden states of RNNs
Query: Like the previous hidden states of the output in RNN network (can be a matrix, two vectors, etc.)
Values: Each value corresponds to one element of the keys

Attention can be confusing because of these three inputs. But:

These inputs are actually often identical. The query is one key and the keys and values are equal. Then if you batch the queries together (one for each key), the n you'll see the query, keys, and values are equal. This is self attention.

Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "**A review on the attention mechanism of deep learning**." *Neurocomputing* 452 (2021): 48-62.

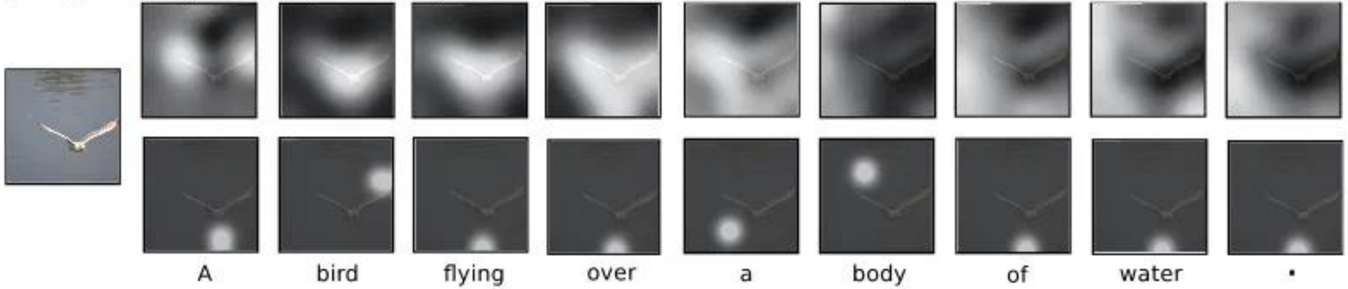
https://github.com/whitead/dmol-book/blob/main/dl/attention.ipynb

Att  ention

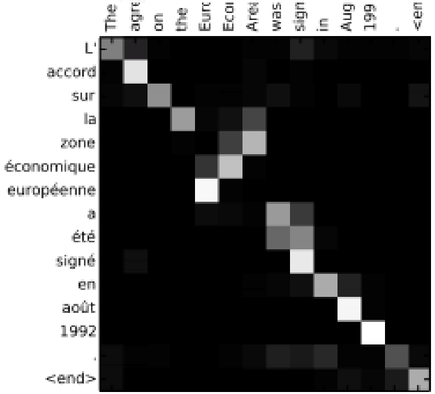
1. Can **improve the performance** of deep learning models, especially in the context of sequential data such as natural language processing (NLP) and speech recognition.
2. The mechanism helps the model **focus on relevant parts of the input data**
3. **Reduces** the amount of **computational resources** required
4. Allows the model to **handle longer sequences of data more efficiently**
5. **Provides a way of interpretability** to the deep learning models, as it provides insight into which parts of the input the model is using to make its predictions.

Interpretability

Figure 2. Attention over time. As the model generates each word, its attention changes to reflect the relevant parts of the image. "soft" (top row) vs "hard" (bottom row) attention. (Note that both models generated the same captions in this example.)



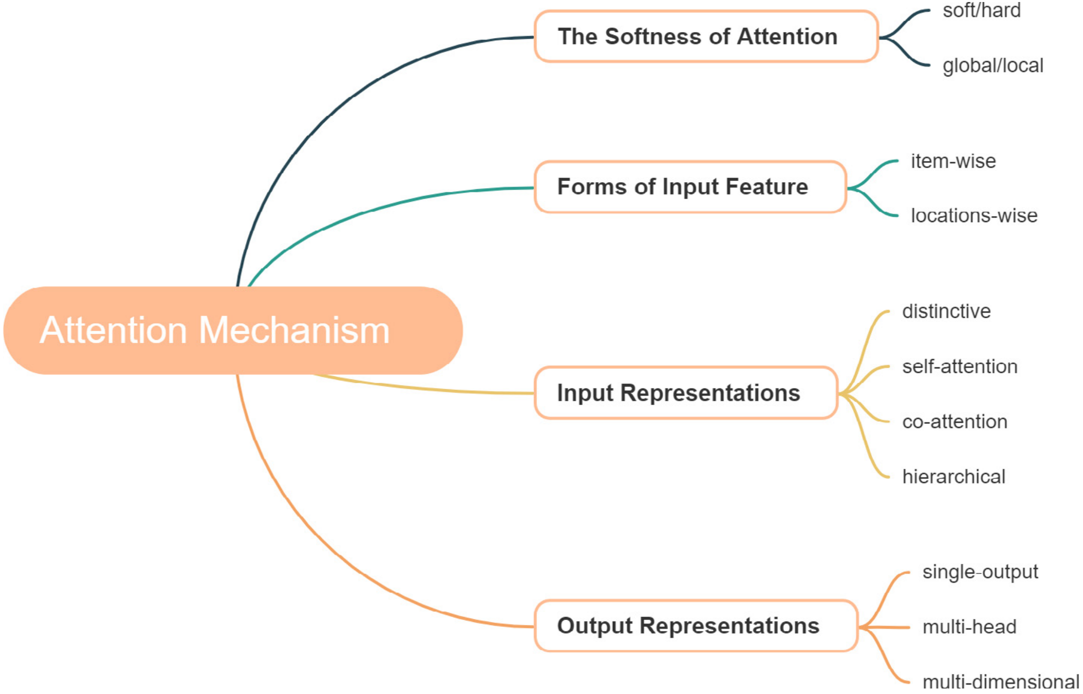
Attention in image captioning
(image to text)



Attention in translation
(text to text)

Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "**A review on the attention mechanism of deep learning**." *Neurocomputing* 452 (2021): 48-62.
https://medium.com/heuritech/attention-mechanism-5aba9a2d4727
Cheng, Jianpeng, Li Dong, and Mirella Lapata. 2016. "Long Short-Term Memory-Networks for Machine Reading." *arXiv Preprint arXiv*:1601.06733.

Attention categories

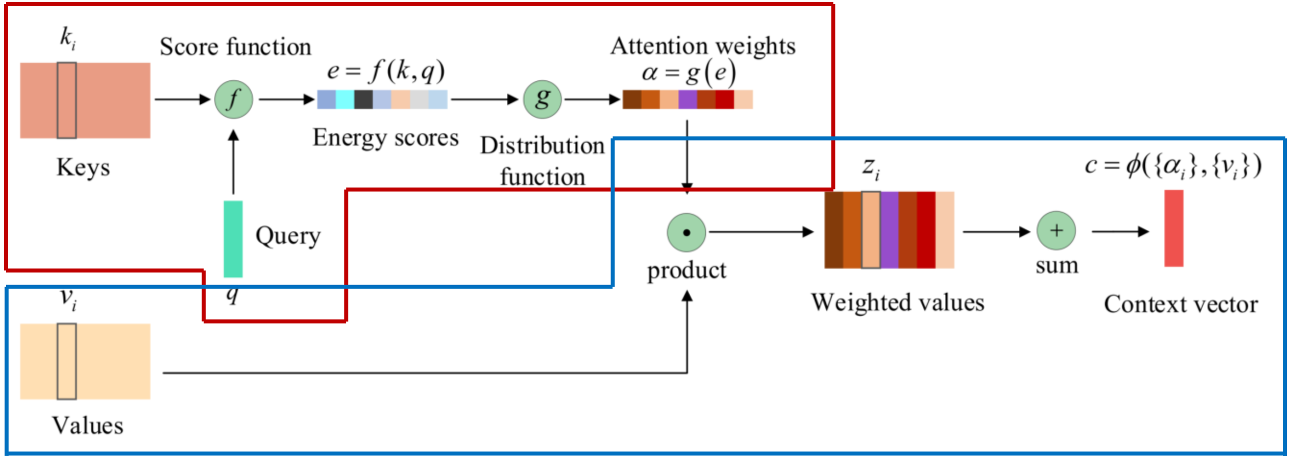


Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "**A review on the attention mechanism of deep learning**." *Neurocomputing* 452 (2021): 48-62.

Computation steps

In simple terms:

1. compute the attention distribution on the input information
2. compute the context vector according to the attention distribution



Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "**A review on the attention mechanism of deep learning**." *Neurocomputing* 452 (2021): 48-62.

Attention mechanism

Compute the score function: the correlation between queries and keys

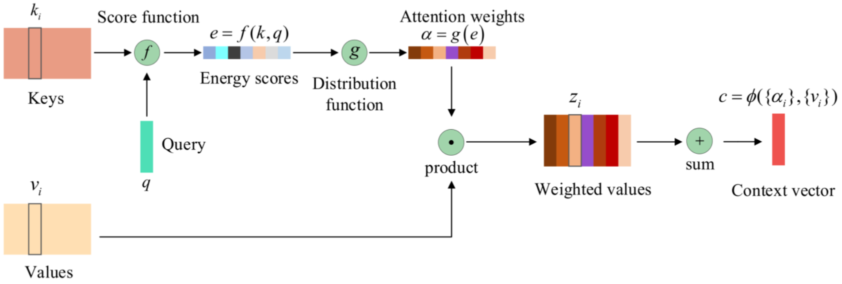
1- Additive
2- multiplicative (dot-product)
3- Scaled multiplicative
4- general
Etc.

Map the score function to attention weights (normalizes all the energy scores to a probability distribution)

1- Softmax
2- Sparsemax
3- Sigmoid
Etc.

Compute the context vector

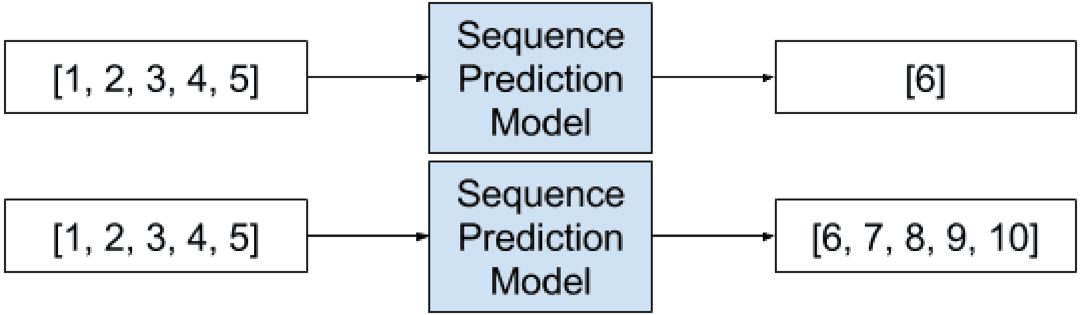
Weighted sum



Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "**A review on the attention mechanism of deep learning**." *Neurocomputing* 452 (2021): 48-62.

Traffic forecasting

Time series forecasting problems



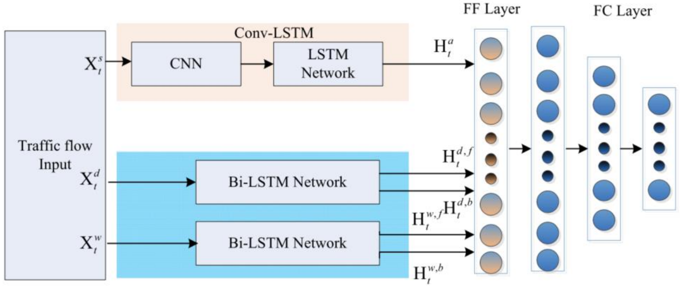
Traffic forecast is a typical time-series prediction problem, i.e. predicting the most likely traffic measurements (e.g. speed or traffic flow) in the next *H* time steps given the previous *M* traffic observations

Traffic forecast is generally classified into two scales: short-term (5 - 30 min), medium and long term (over 30 min).

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun et al. "**Google's neural machine translation system: Bridging the gap between human and machine translation**." *arXiv preprint arXiv*:1609.08144 (2016).

Case study

A Hybrid Deep Learning Model With Attention-Based Conv-LSTM Networks for Short-Term Traffic Flow Prediction



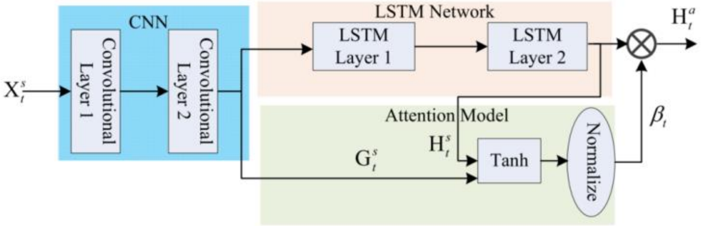


Fig. 2. The Conv-LSTM module with an attention mechanism.

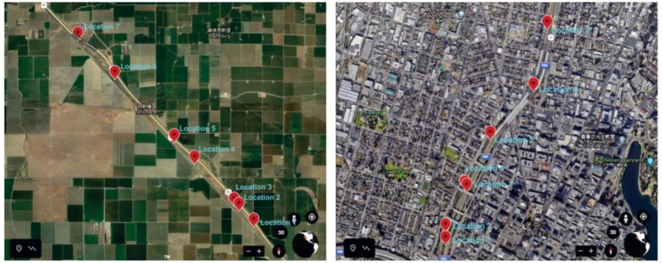
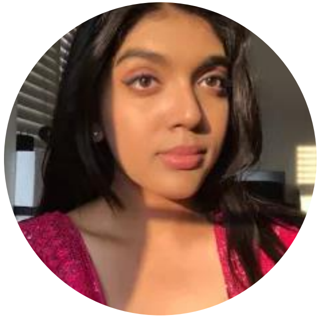
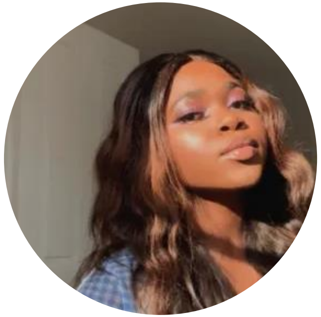


Fig. 6. Sensor distribution on Freeway SR99-S (left) and Street I980 (right).

Zheng, Haifeng, Feng Lin, Xinxin Feng, and Youjia Chen. "**A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction**." *IEEE Transactions on Intelligent Transportation Systems* 22, no. 11 (**2020**): 6910-6920.
https://github.com/suprobe/AT-Conv-LSTM

CAIS





Farzad Roozitalab

PhD Student and research assistant
CAIS workshop lead