RNN & GNN

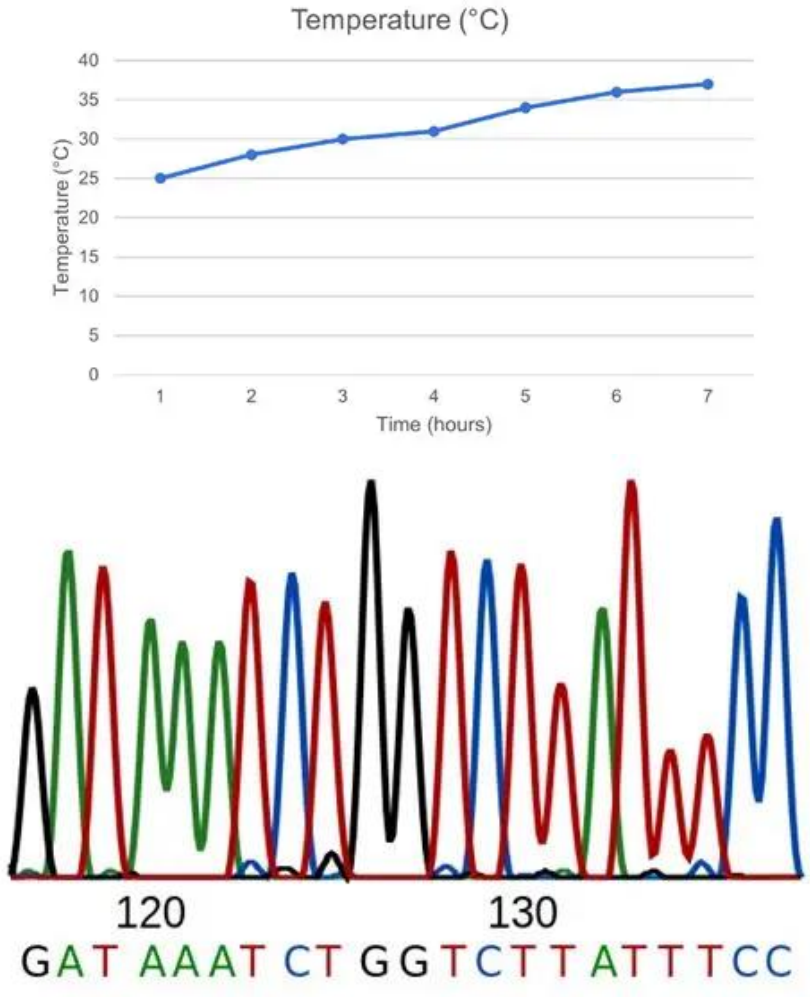
Weihao Zhao

Mar. 2, 2026

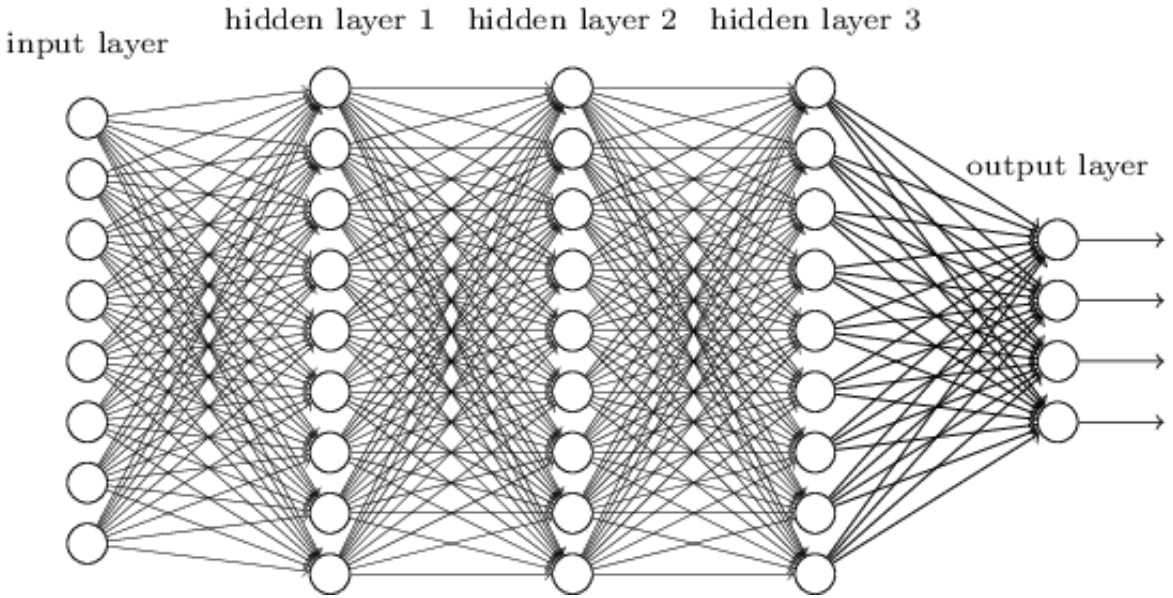
Contents

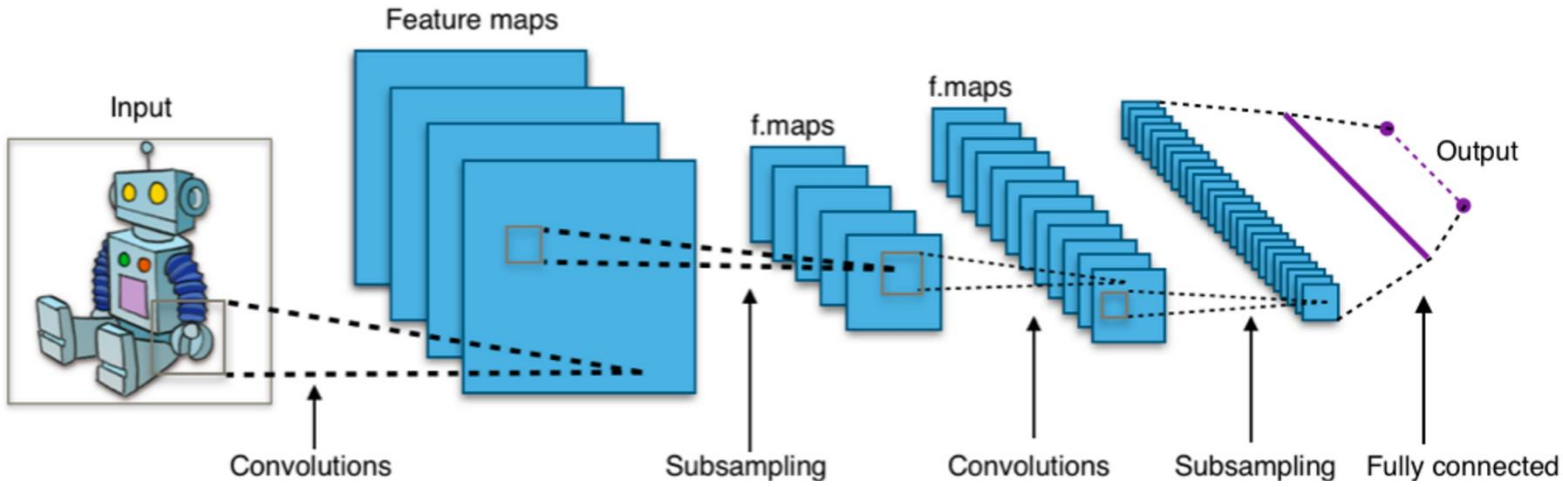
Recurrent Neural Networks (RNN)

Sequence data with variable length is abundant in real world

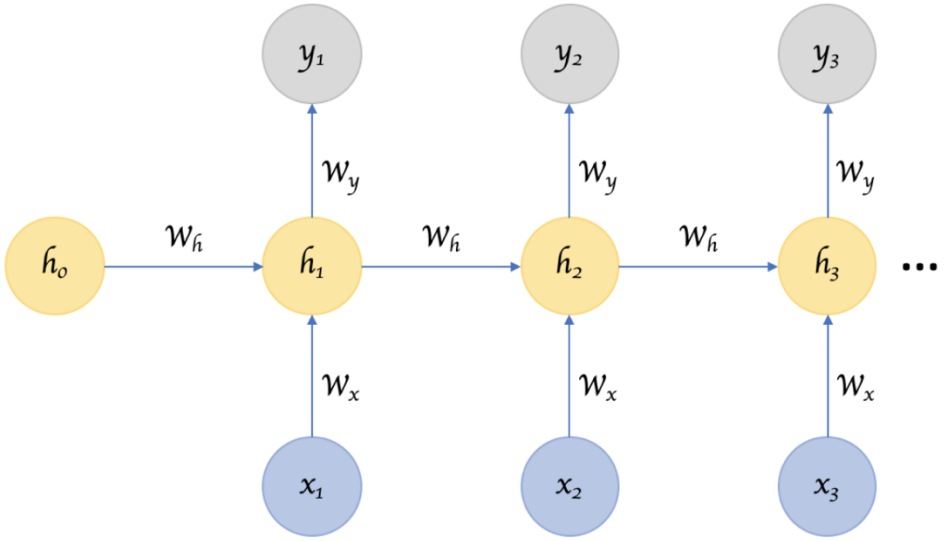


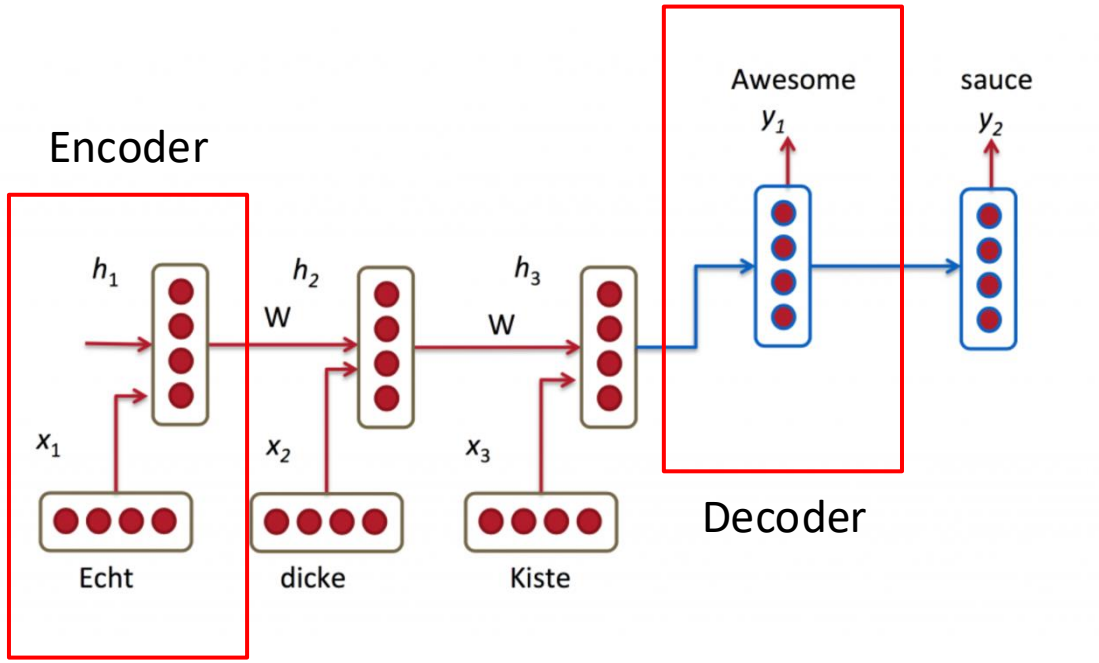
Standard NN and CNN could not deal with input with variable length





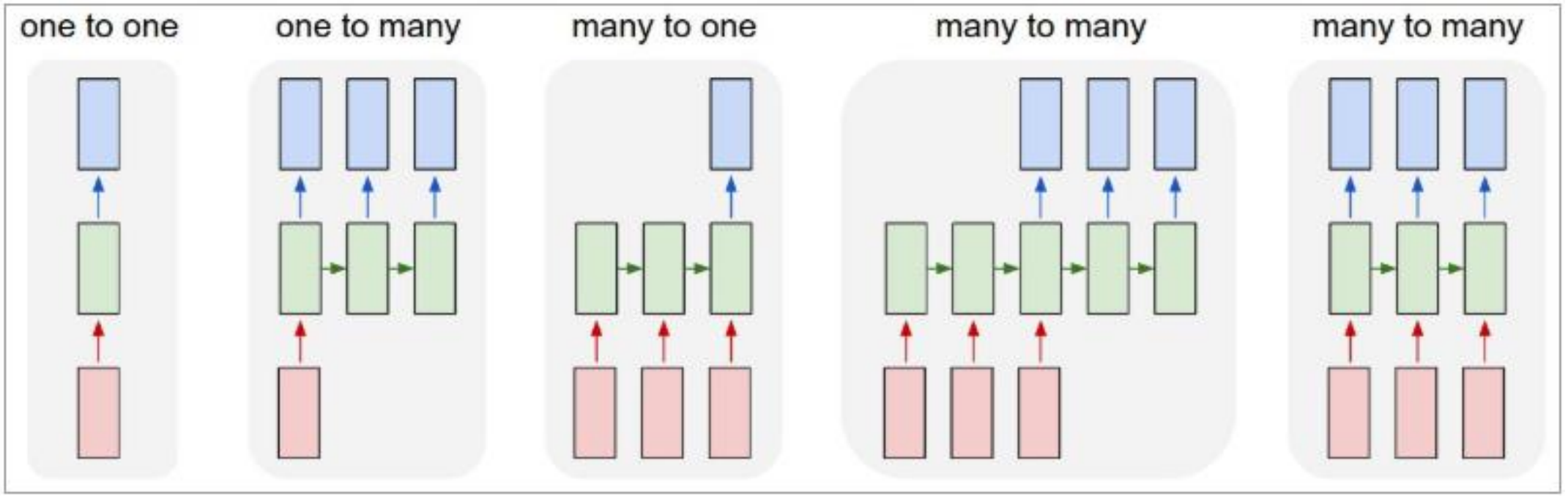
Recurrent Neural Networks could process sequence data with memory to previous parts





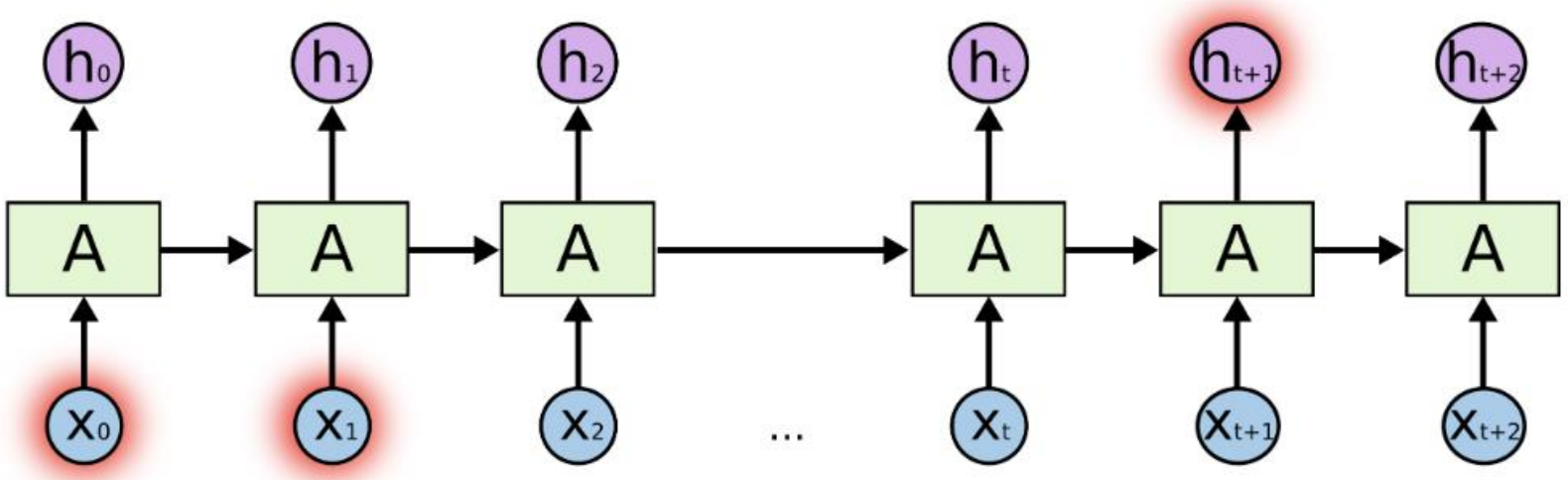
https://iq.opengenus.org/types-of-neural-networks/
https://blog.csdn.net/mwcxz/article/details/129600591

RNN is capable of multiple kinds of sequence-oriented tasks



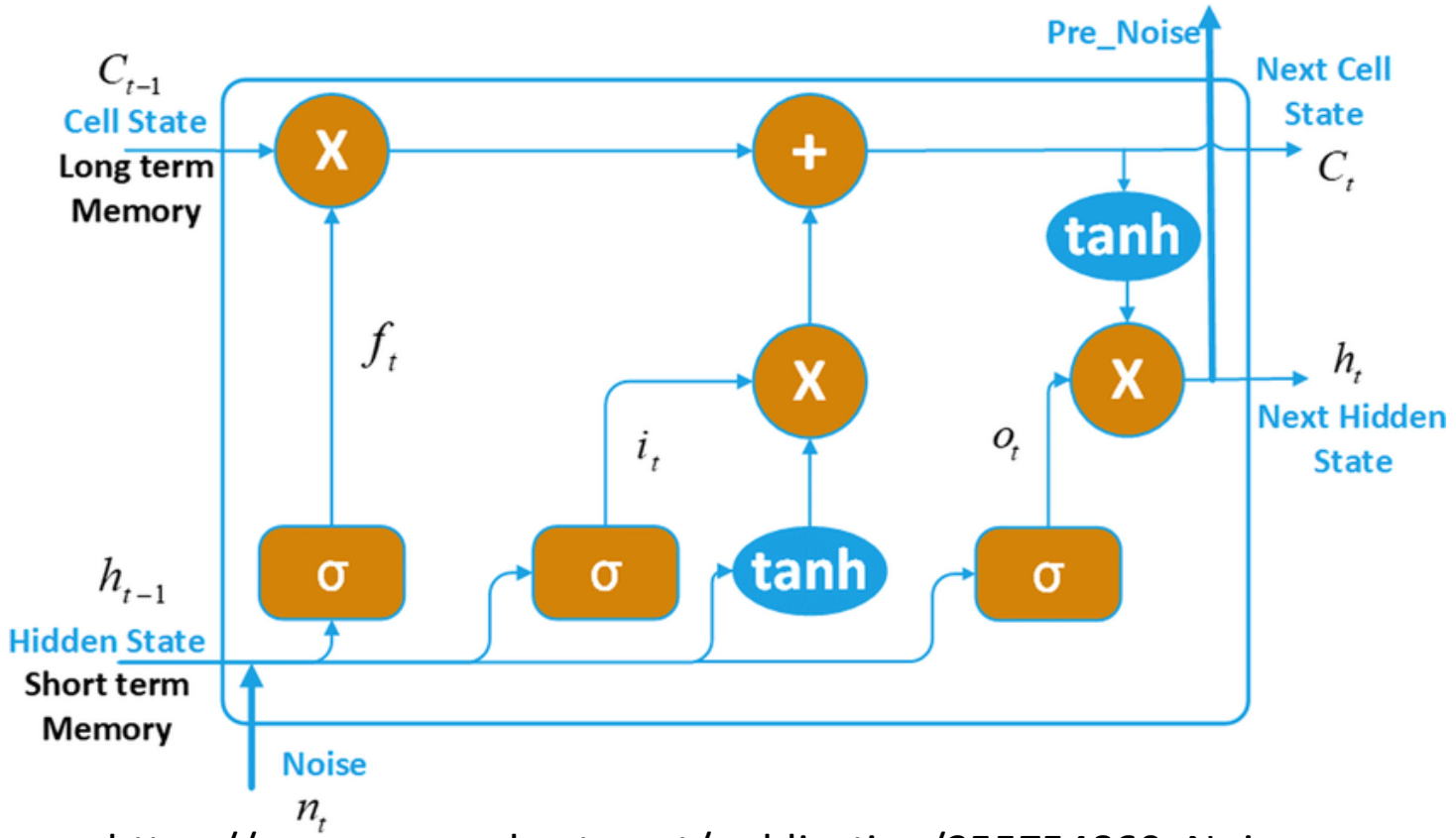
https://www.researchgate.net/publication/331000984_Predictive_Fusion_Model_for_Multi-Modal_Data_Streams

Vanila RNN might fail to remember long term relationship



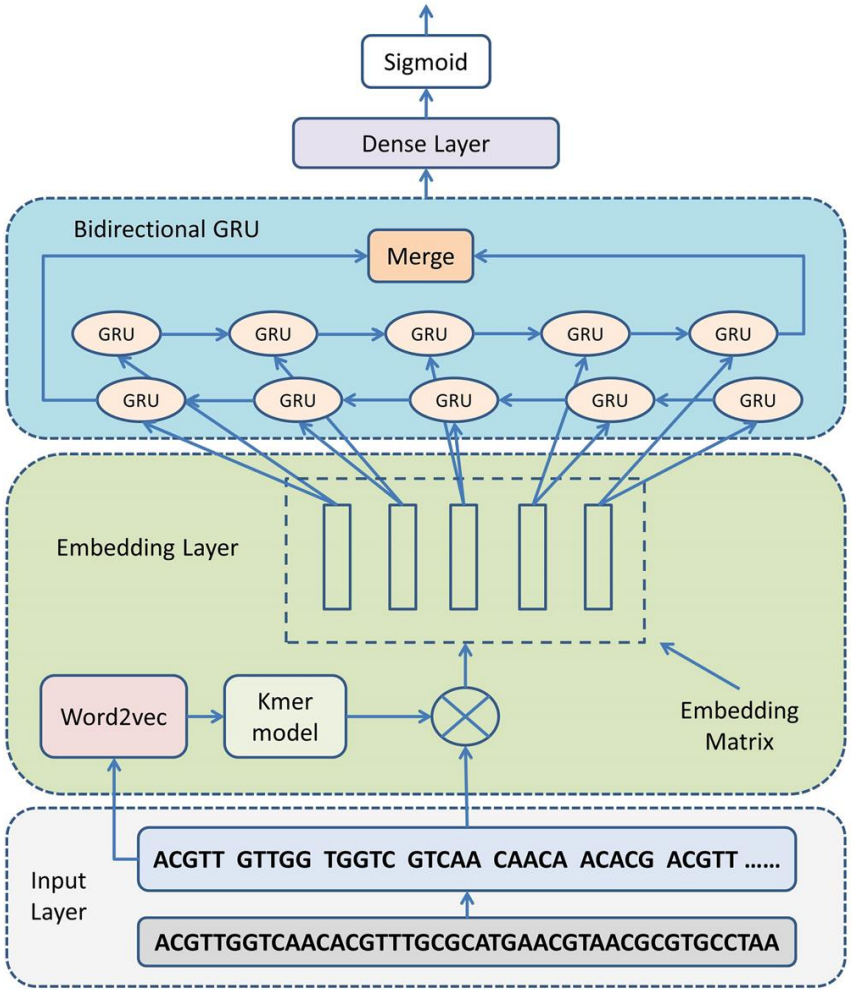
https://colah.github.io/posts/2015-08-Understanding-LSTMs/

LSTM (Long/Short-Term Memory) memorizes information from inputs no matter near or far away



https://www.researchgate.net/publication/355754860_Noise_prediction_of_chemical_industry_park_based_on_multi-station_Prophet_and_multivariate_LSTM_fitting_model

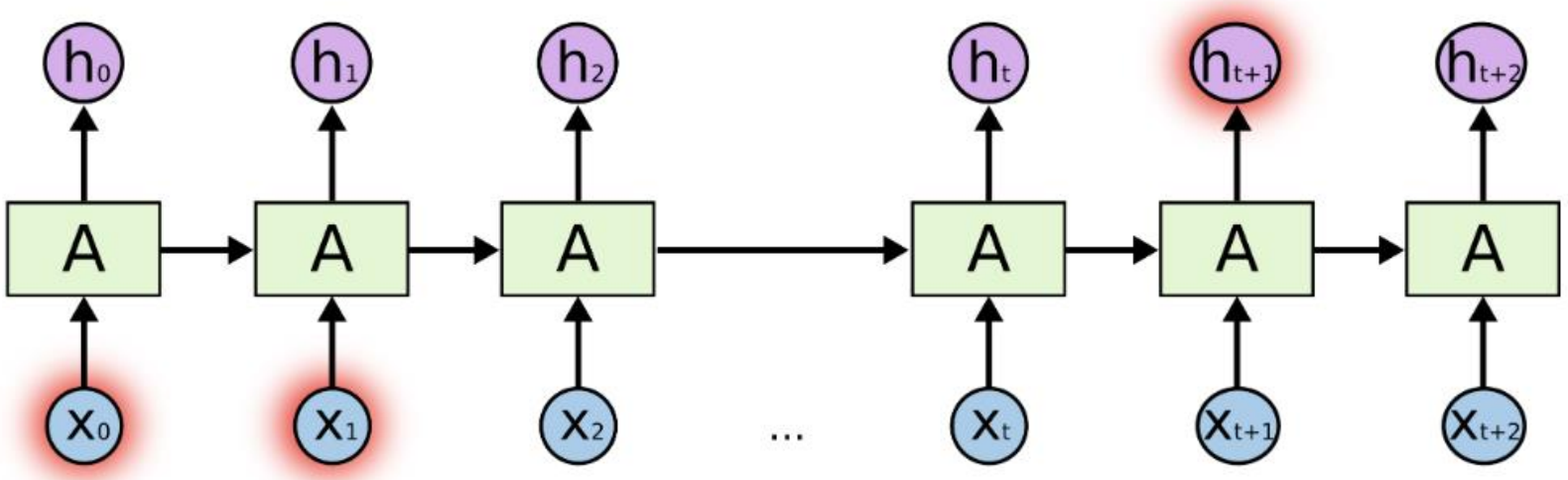
Application of RNN on DNA sequence



Shen et al., **Sci Rep**. 2018

Attention & Transformers

Another way of solving the problem with RNN?



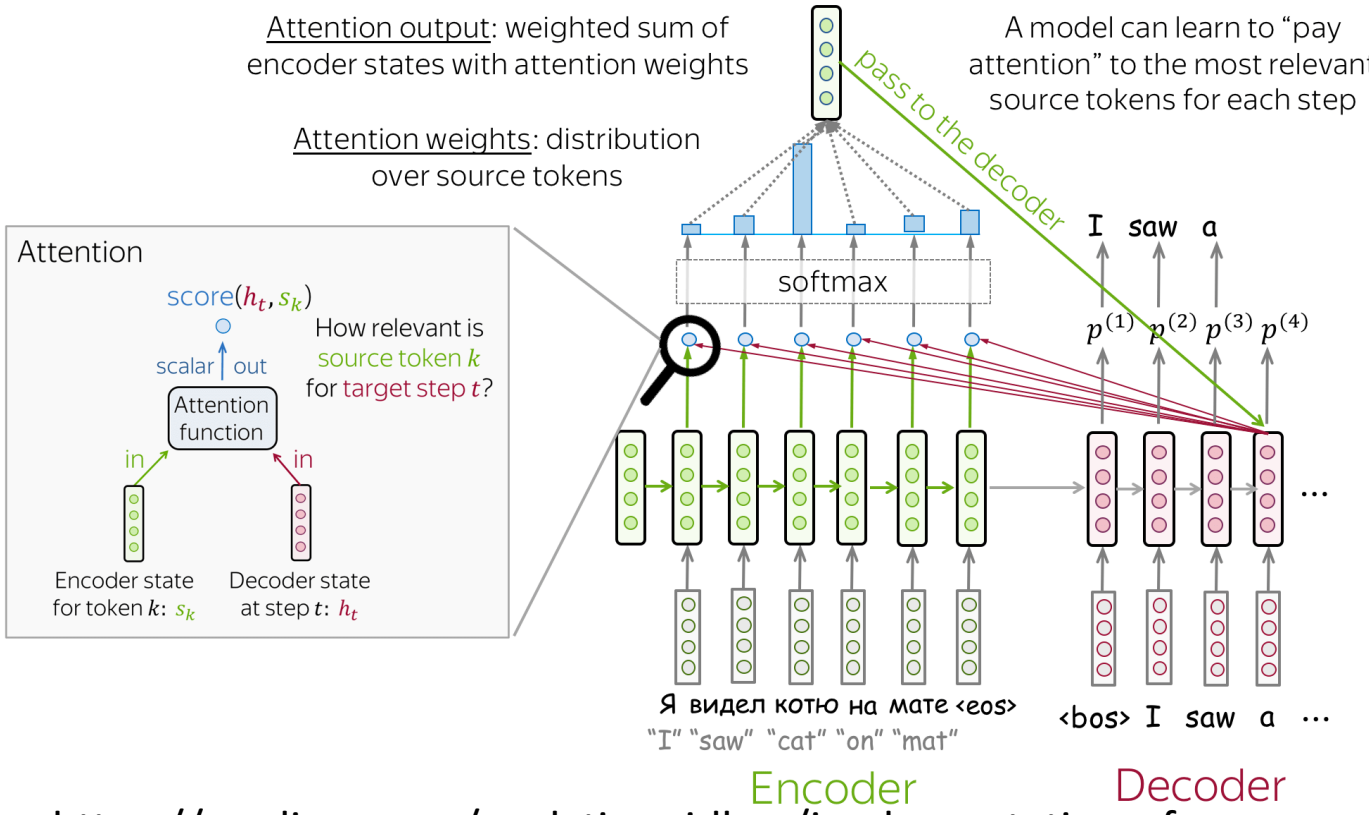
All input sequence ”tokens” are the same in vanilla RNN

– Can we allow the model to selectively choose which part of the sequence it want?

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

Define attention to address the long-range dependencies in RNN

- Attention: a technique used in deep learning that allows the model to selectively focus on specific areas of the input data when making predictions.



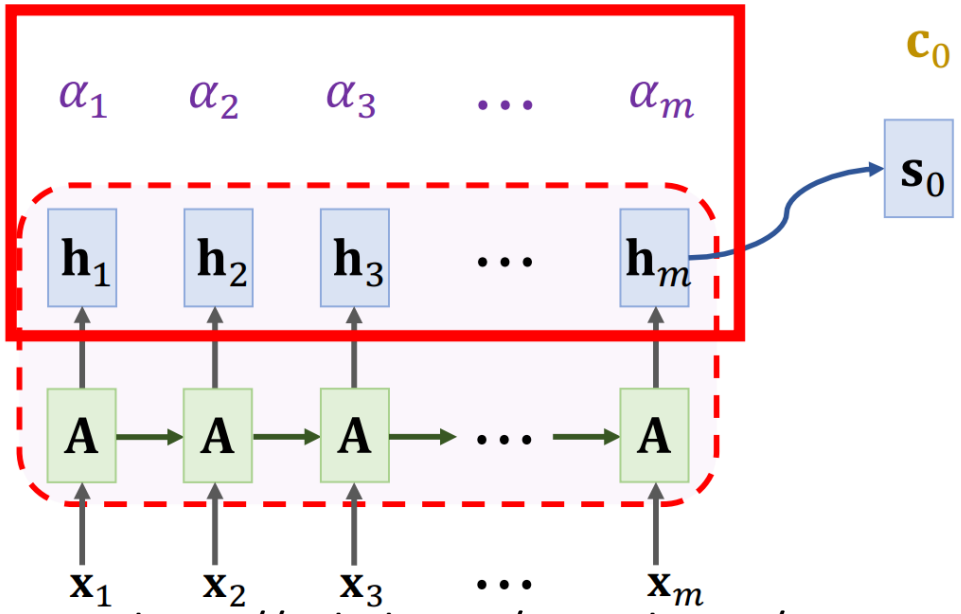
https://medium.com/analytics-vidhya/implementation-of-neural-machine-translation-using-attentions-3b36337a5b23

Calculation of attention in RNN

SimpleRNN + Attention

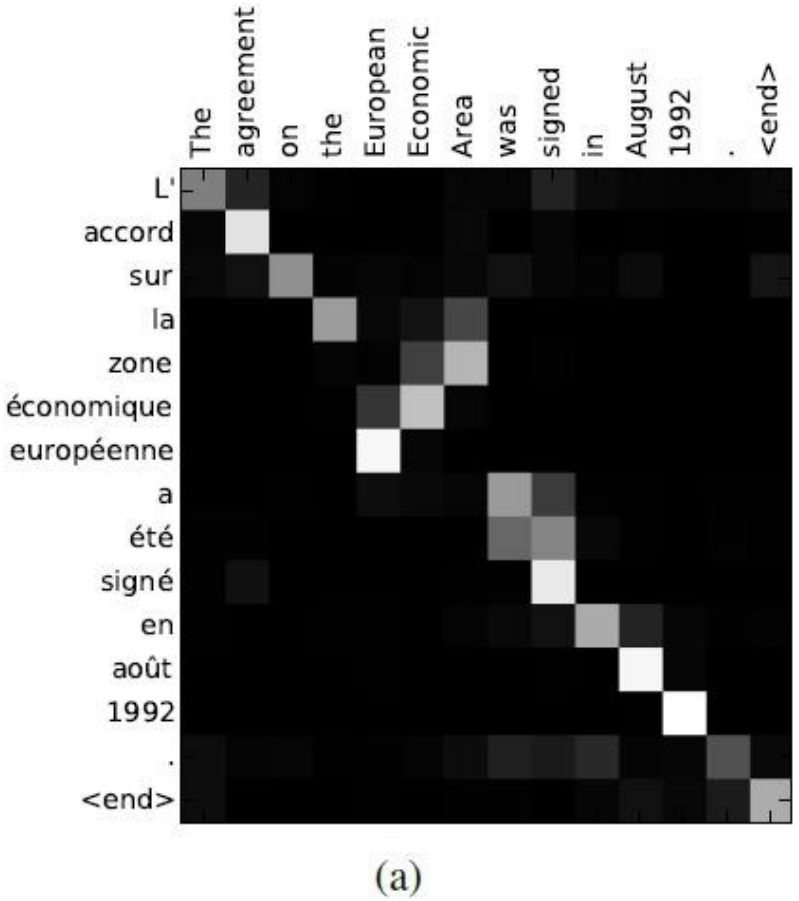
Weight: $\alpha_i = \text{align}(\mathbf{h}_i, \mathbf{s}_0)$.

Context vector: $\mathbf{c}_0 = \alpha_1 \mathbf{h}_1 + \cdots + \alpha_m \mathbf{h}_m$.



https://github.com/wangshusen/DeepLearning/blob/master/Slides/9_RNN_8.pdf

Attention can also help explain the model



(a)

https://arxiv.org/abs/1409.0473

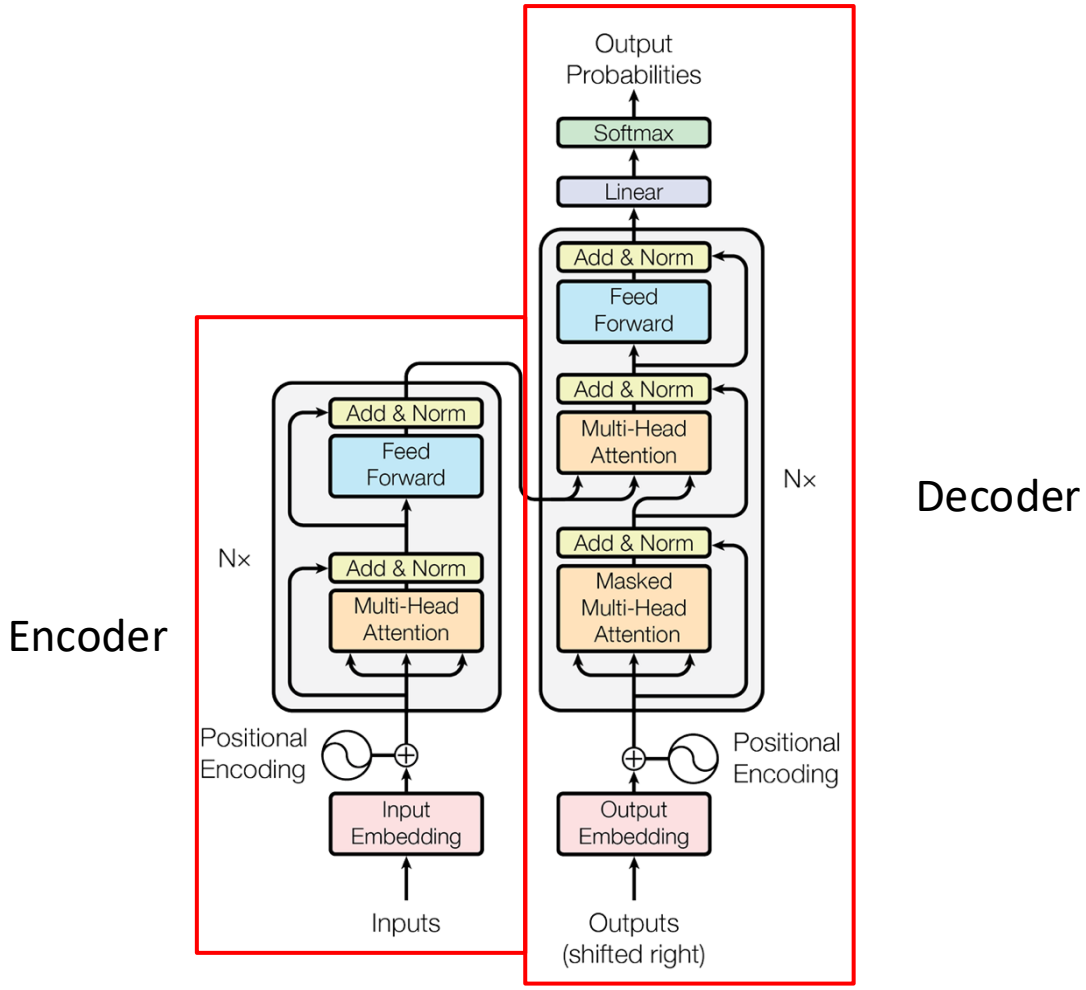
Actually… attention is all you need

- The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. (Vaswani et al., 2017)

- Self-attention: an attention mechanism relating different positions of a single sequence in order to compute a representation of the sequence

- Embedding: a way to represent discrete objects (like words, images, audio, or graph nodes) as dense numerical vectors in a continuous space

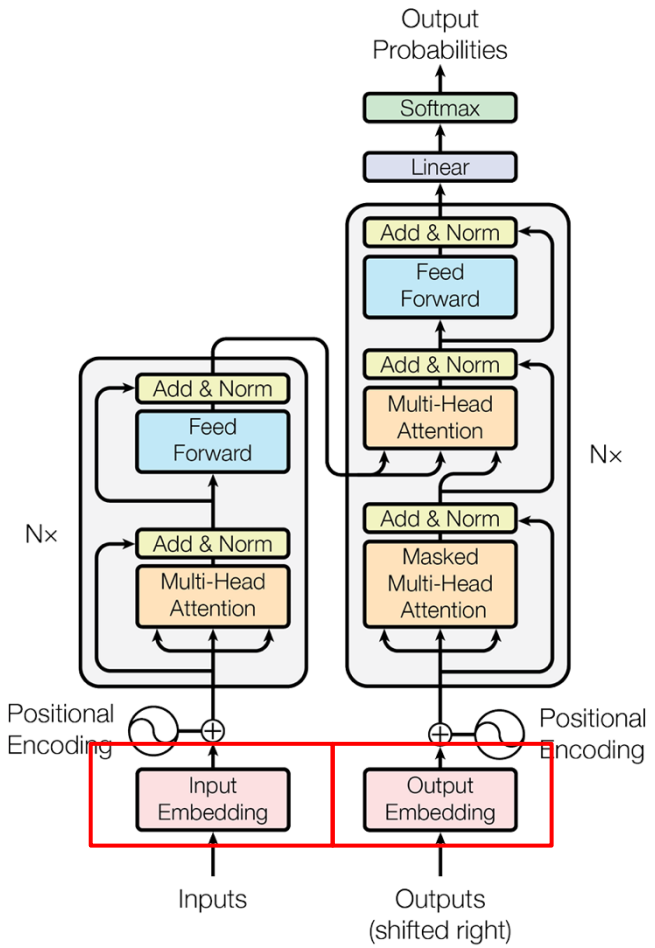
Overall architecture of transformers



https://arxiv.org/abs/1706.03762

Embedding for input and output sequence

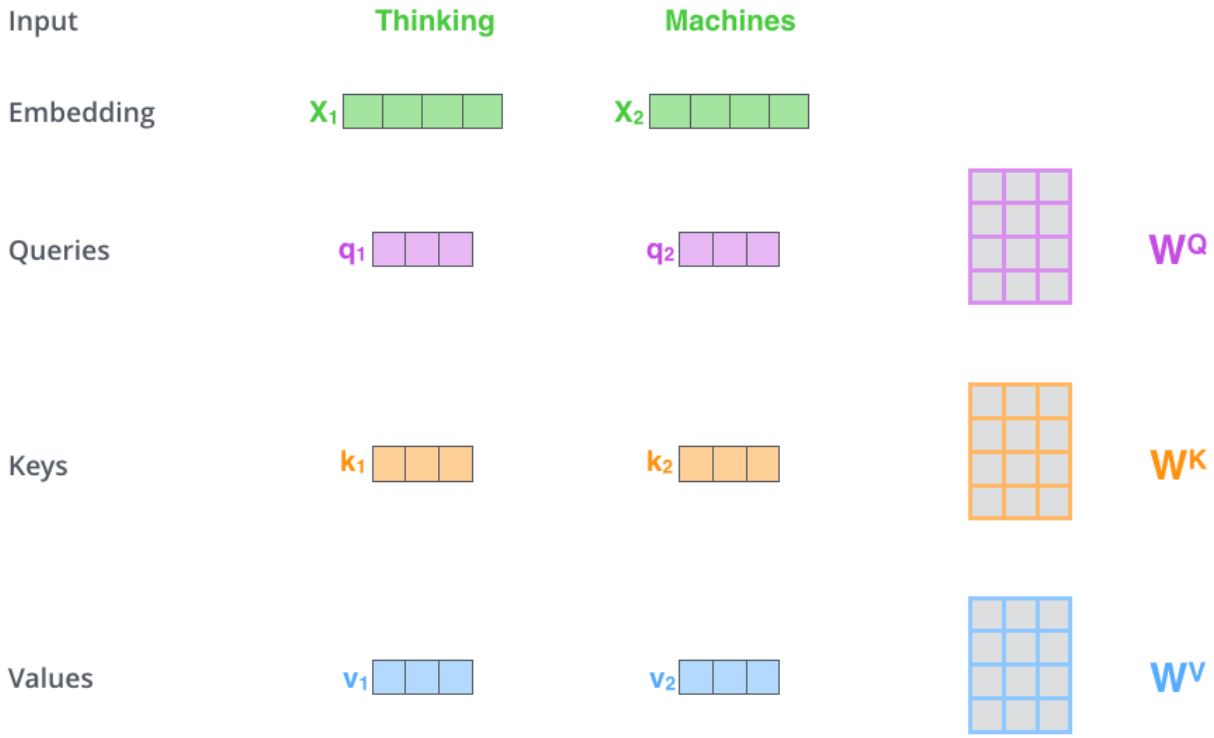
- Embedding: produce a vector for each input/output token
- Concatenated with positional encoding (produce a value for each token's place in the sequence), this becomes the input to the Multi-Head Attention



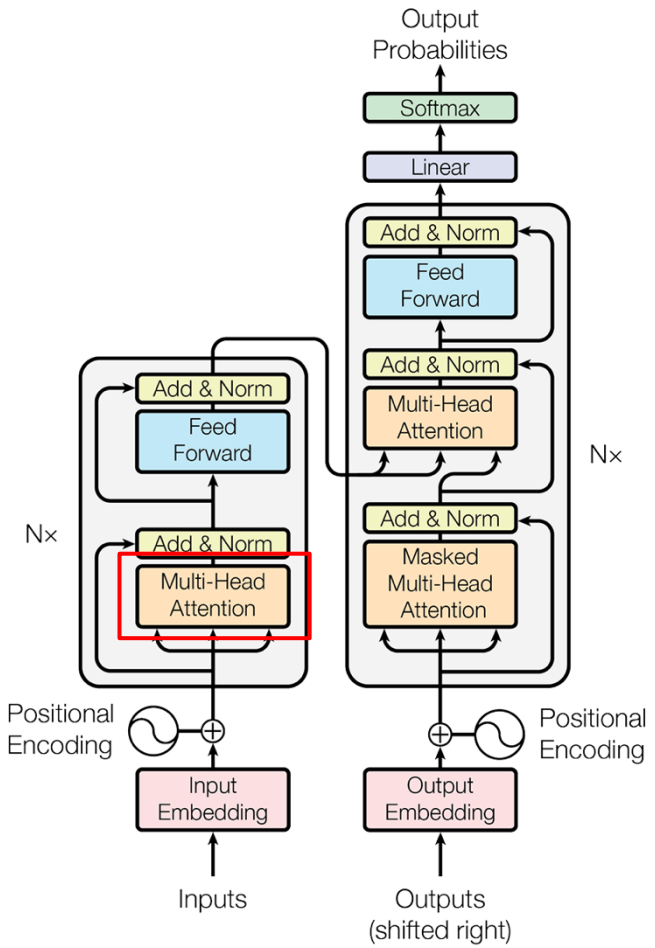
https://arxiv.org/abs/1706.03762

Multi-Head Self-Attention

- Attention is calculated by first projecting the input embedding as query, key, and value



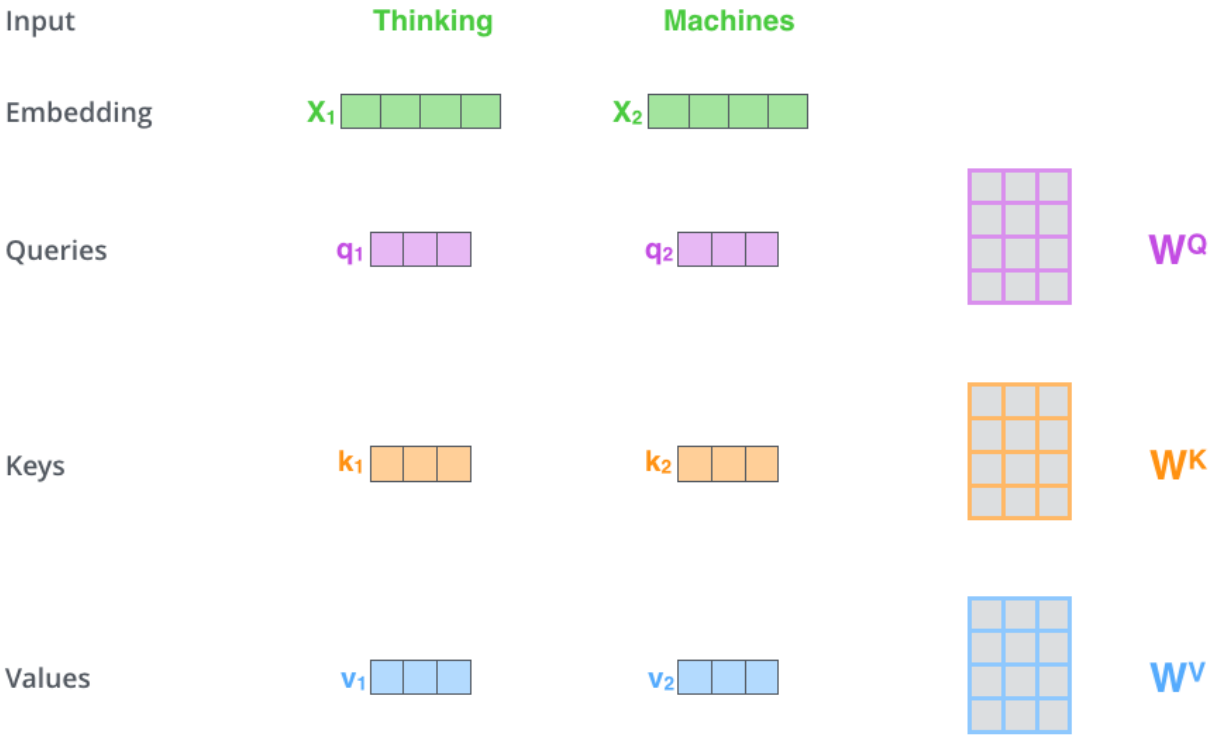
https://blog.csdn.net/2301_82275412/article/details/147088123

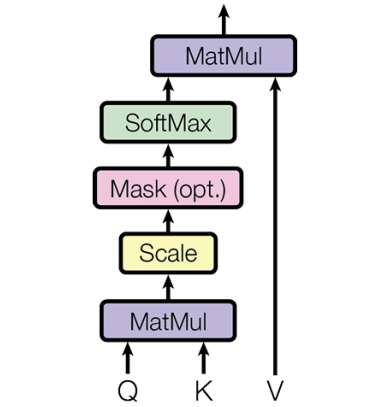


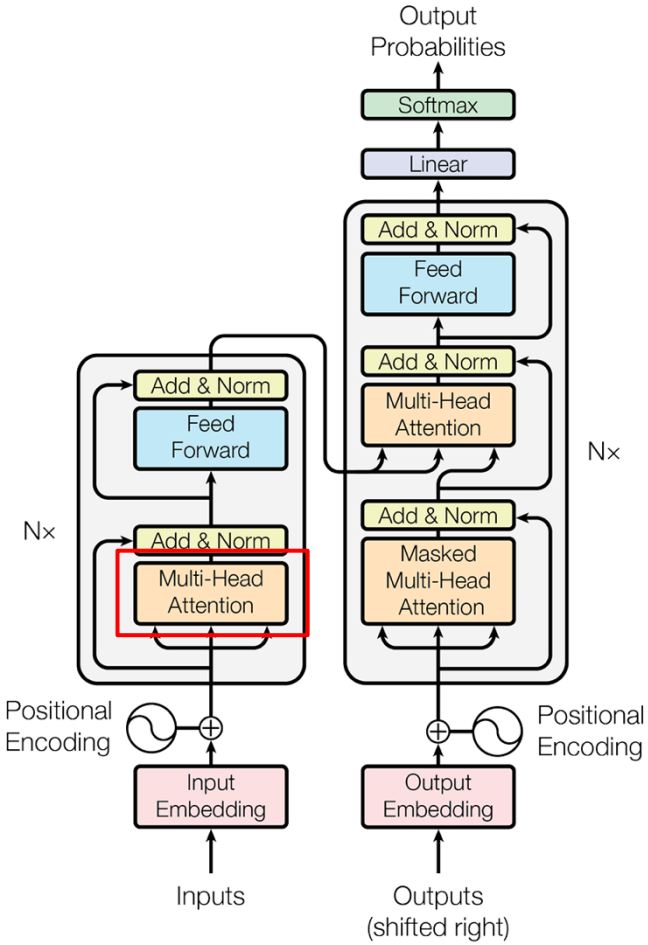
https://arxiv.org/abs/1706.03762

Multi-Head Self-Attention

- After we have the query, key, and value, attention is calculated as: $\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$





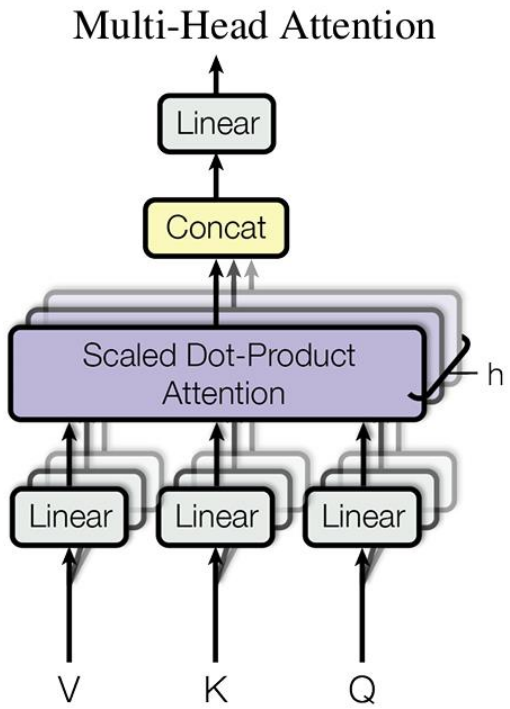


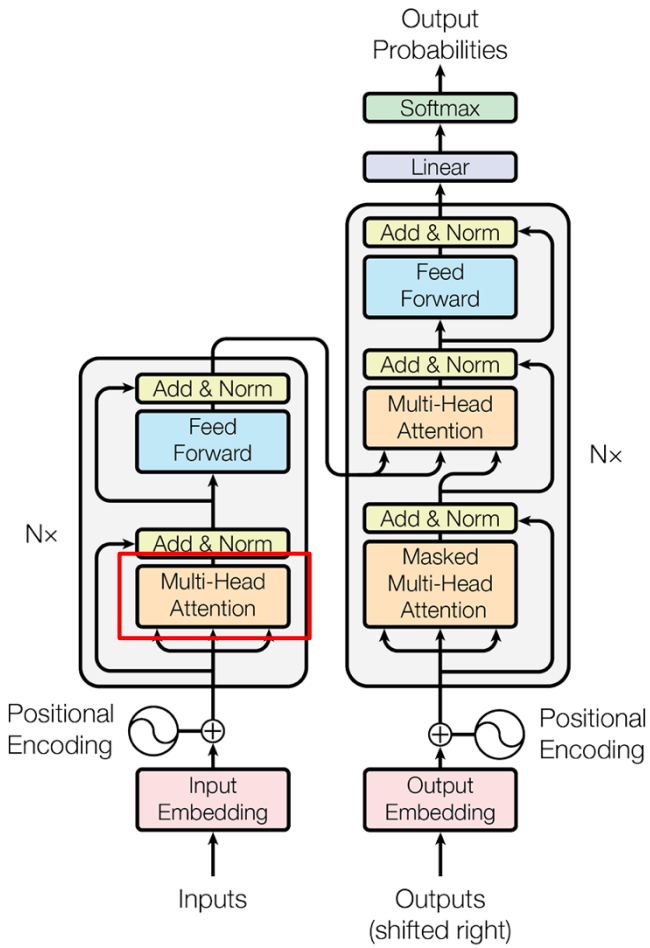
https://blog.csdn.net/2301_82275412/article/details/147088123

https://arxiv.org/abs/1706.03762

Multi-Head Self-Attention

- Multi-Head means that the attention calculation will happen many times in parallel, catching different levels of context.

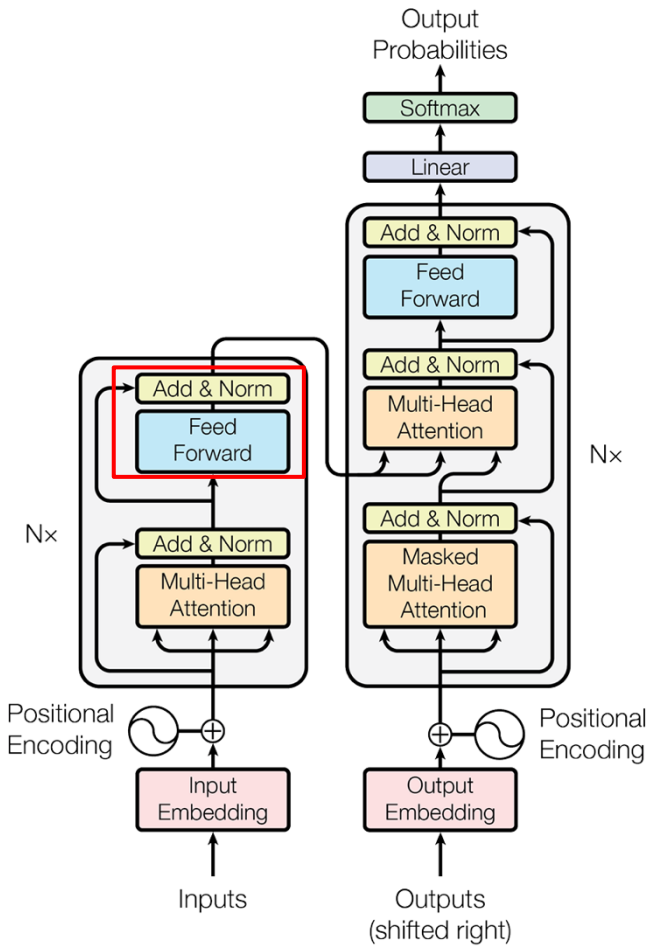




https://arxiv.org/abs/1706.03762

Feed forward layers and normalization

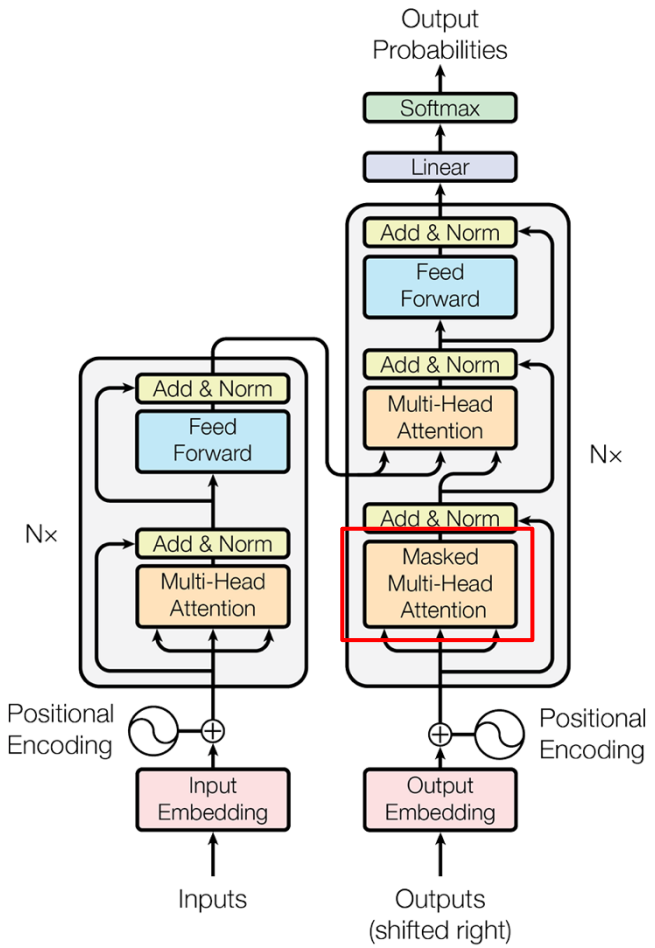
- Add more linear layers to the attention matrix, then combine with the original attention, normalize to unit variance



https://arxiv.org/abs/1706.03762

Masked Multi-Head Self-Attention

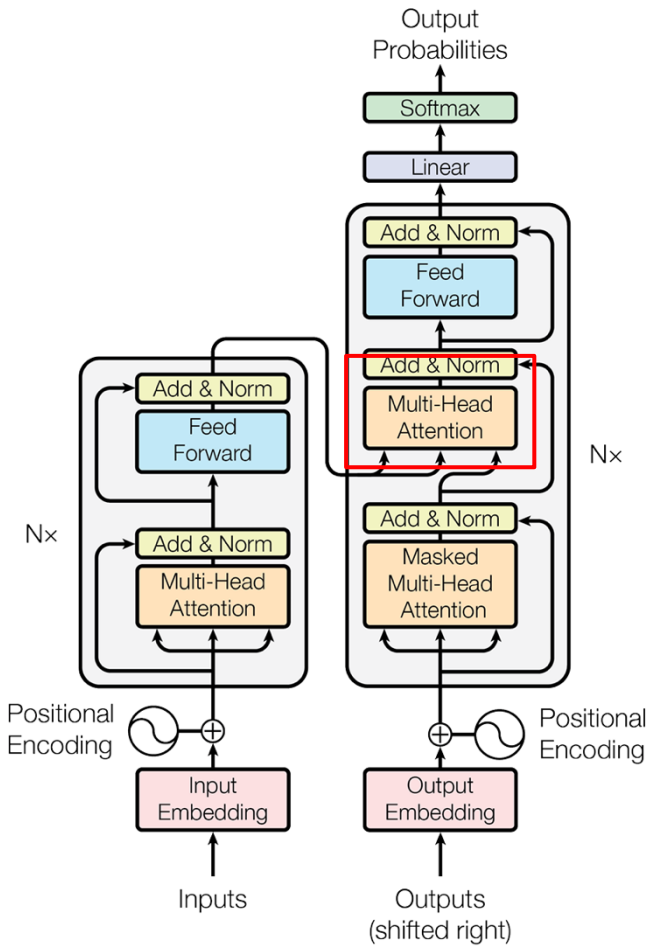
- Similar to previous Multi-Head Self-Attention, but will mask future tokens to avoid peeking



https://arxiv.org/abs/1706.03762

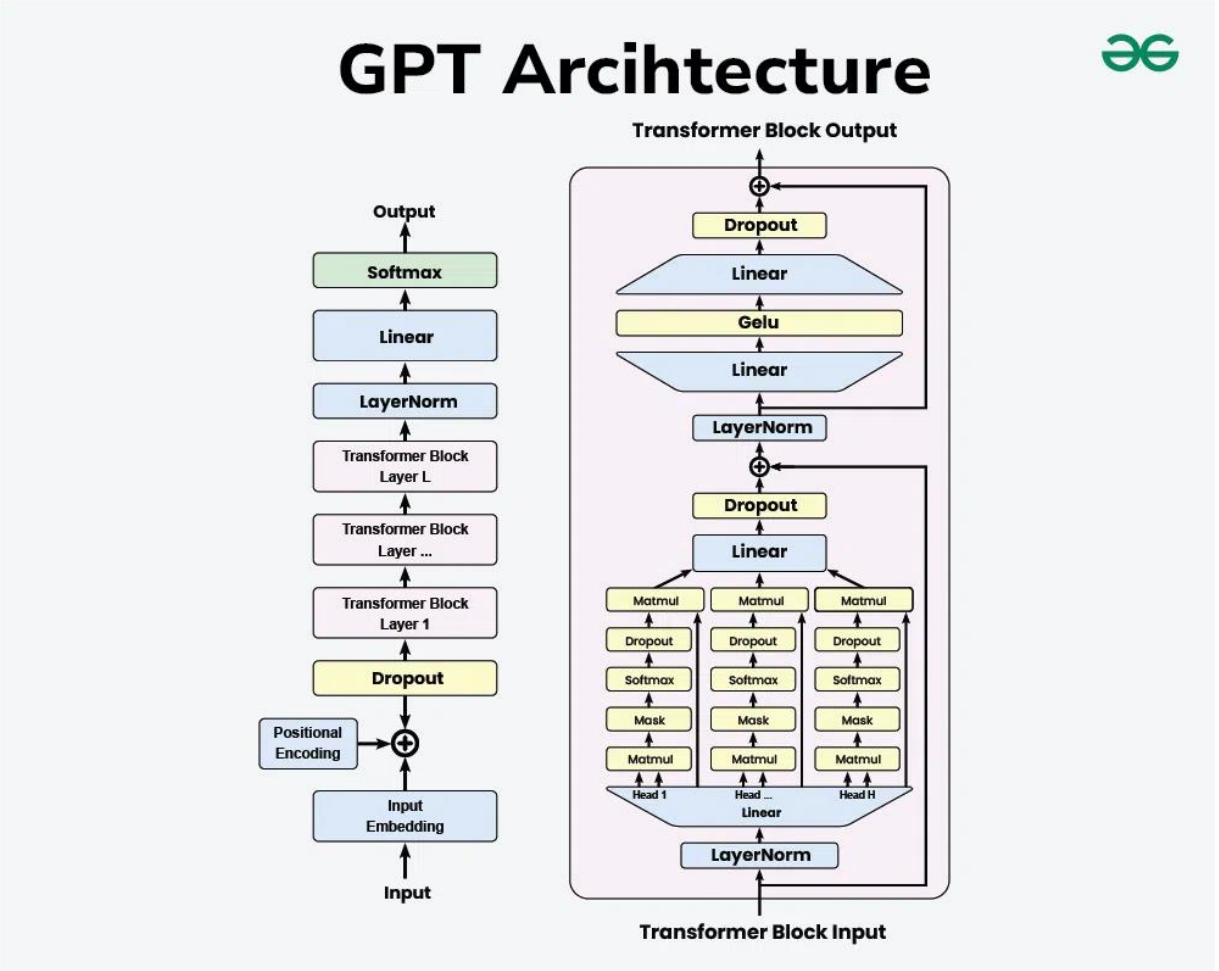
Multi-Head Attention in decoder layer

- The query comes from the decoder itself or the previous decoder layer
- The key and value comes from the encoder layer
- Output probability of the next token



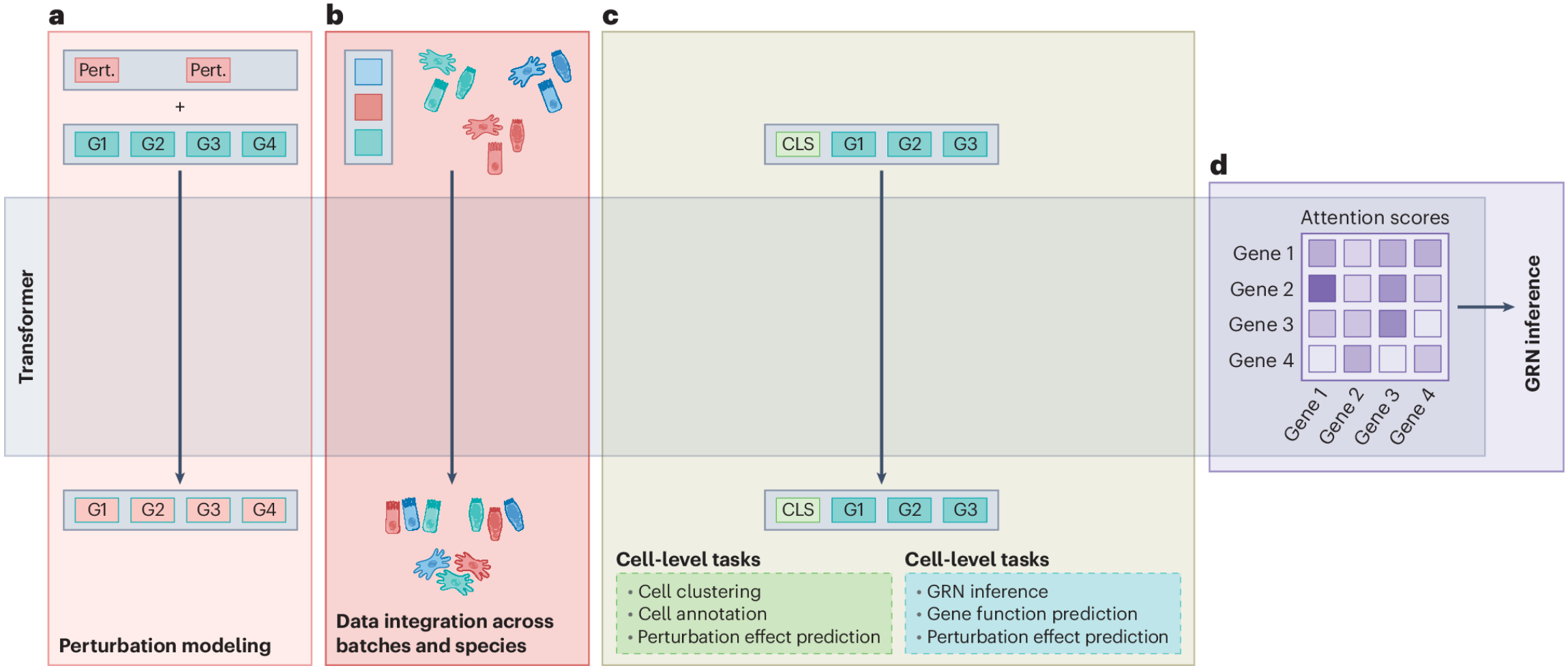
https://arxiv.org/abs/1706.03762

Transformer is the foundation of current LLM and AI wave



https://yozm.wishket.com/magazine/detail/2696/

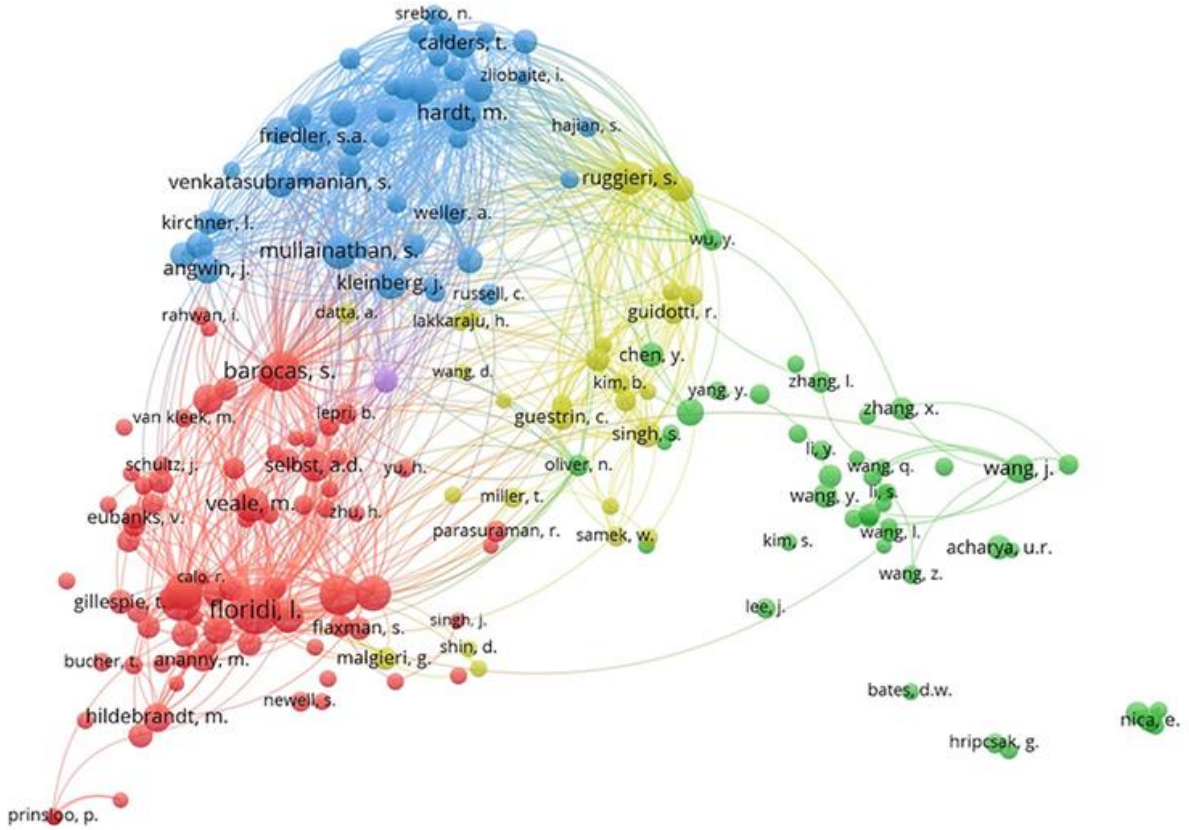
Application of transformer in bioinformatics

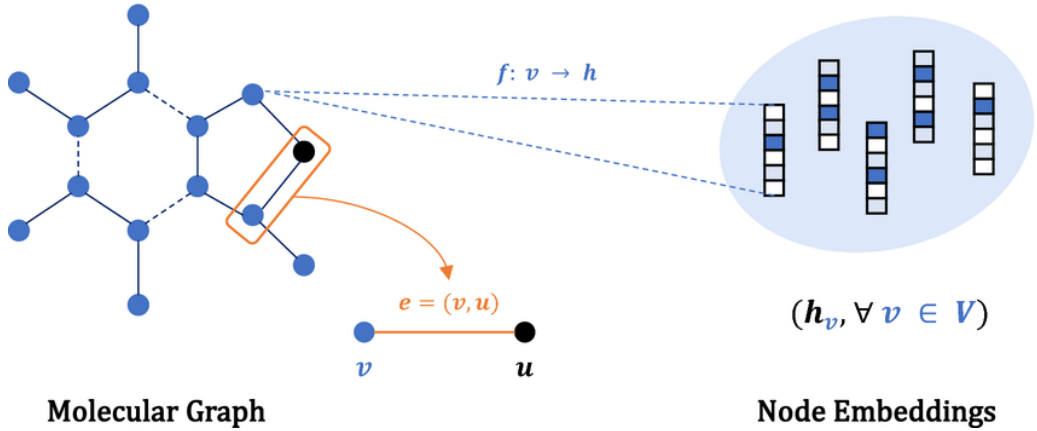


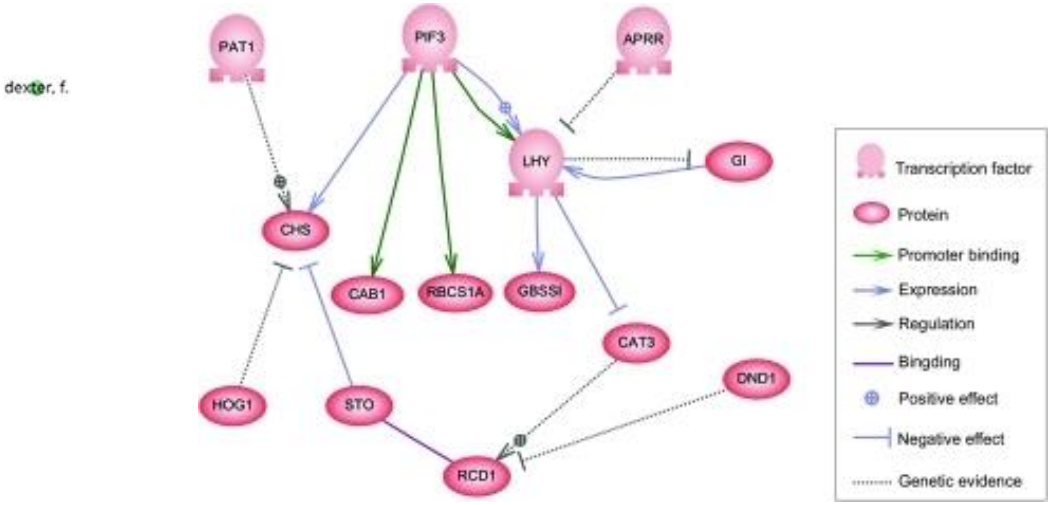
Szałata et al., **Nat. Methods**. 2024

Graph Neural Networks (GNN)

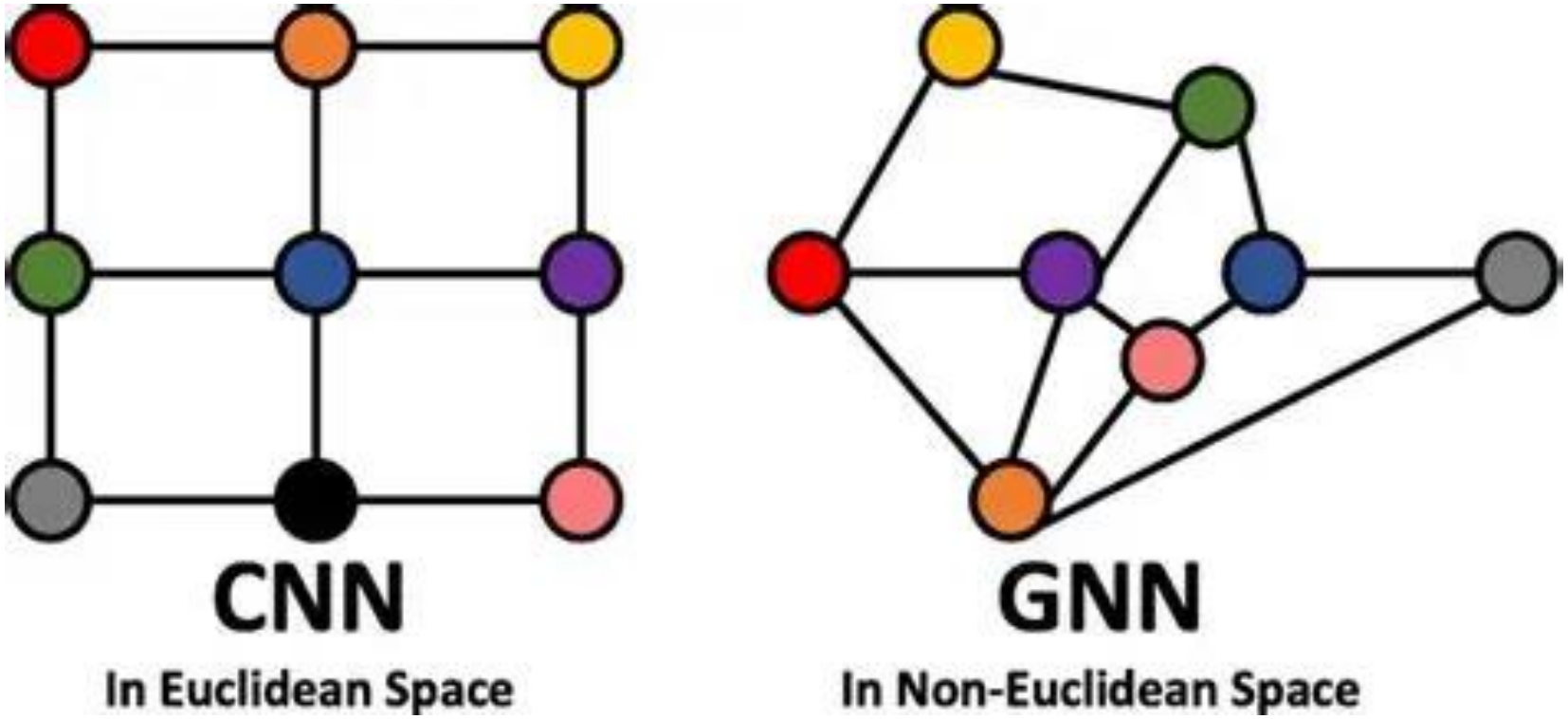
Some real-world data exists as networks





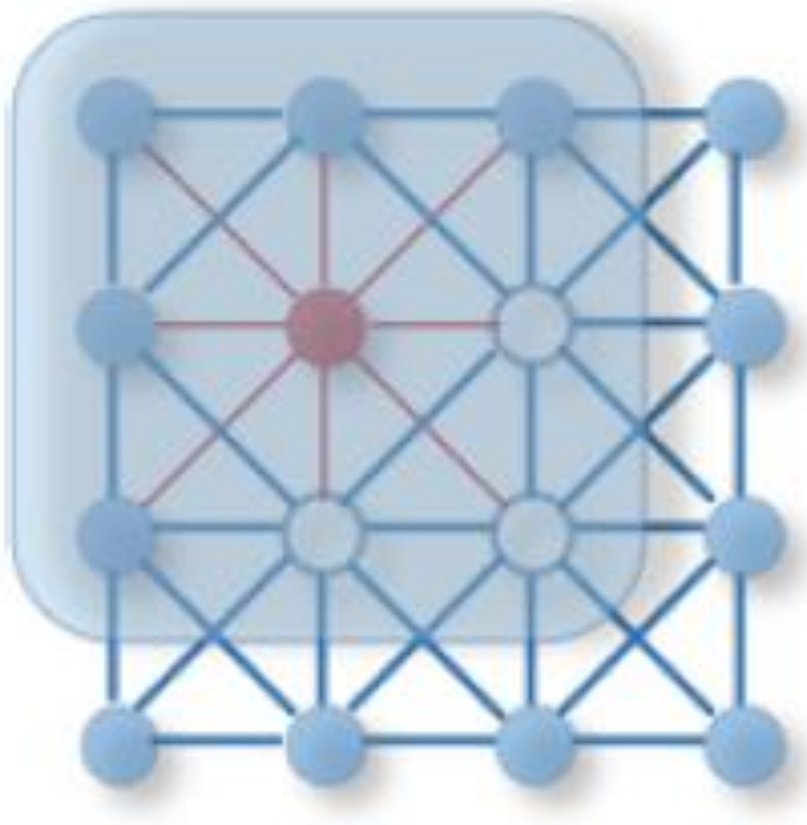


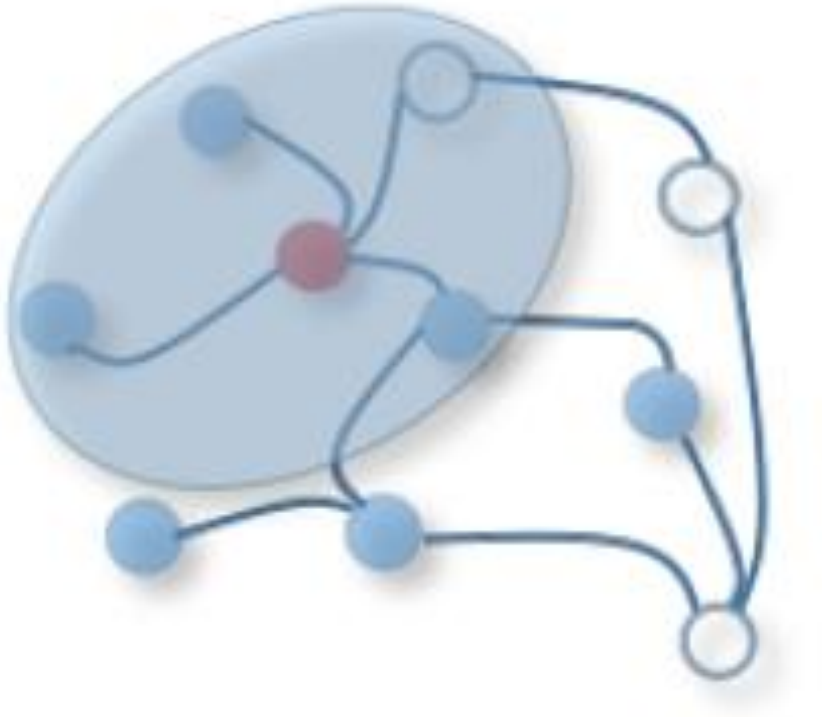
CNN is actually a special form of GNN



https://www.researchgate.net/publication/343821039_A_Survey_on_Deep_Learning-Based_Vehicular_Communication_Applications

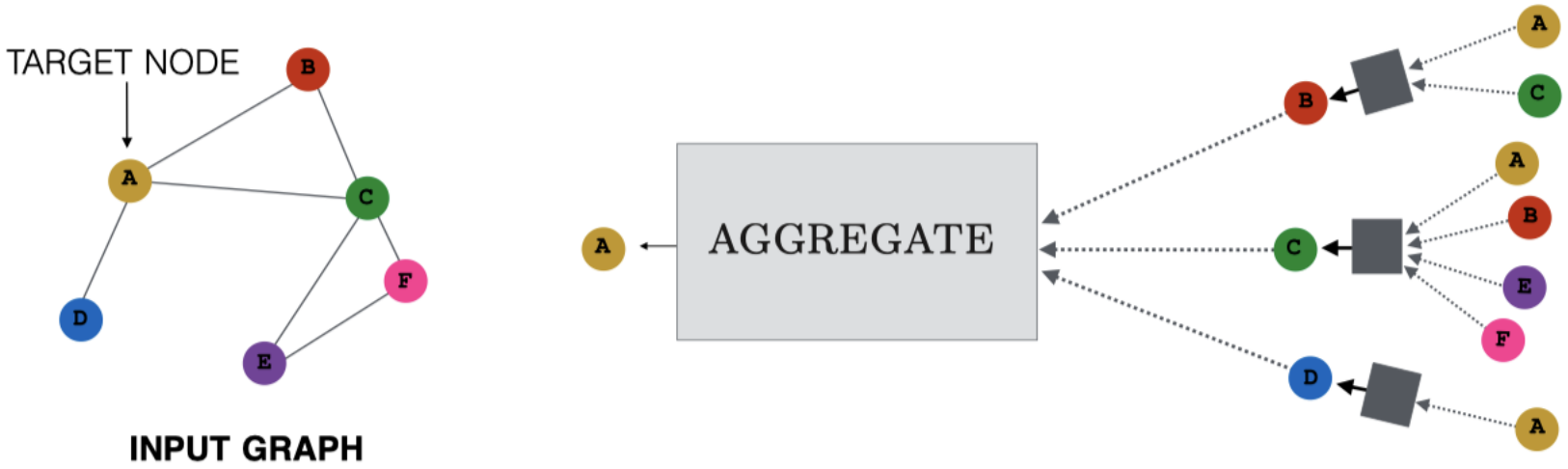
Design a new convolution layer for graphs





https://ichi.pro/setsumei-kanona-gurafunyu-rarunettowa-ku-ni-mukete-76922387246544

Convolution through aggregation



$$\mathbf{h}_u^{(k+1)} = \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)}(\{\mathbf{h}_v^{(k)}, \forall v \in \mathcal{N}(u)\})\right)$$
$$= \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \mathbf{m}_{\mathcal{N}(u)}^{(k)}\right),$$

https://www.researchgate.net/publication/364071895_An_Integer_Programming_Approach_Reinforced_by_a_Message-passing_Procedure_for_Detecting_Dense_Attributed_Subgraphs

Graph attention in aggregation

- We can also apply attention mechanism to the mean function used to aggregate neighboring information to allow the model selectively focus on specific nodes

$$e_{vu} = a(\mathbf{W}^{(l)}\mathbf{h}_u^{(l-1)}, \mathbf{W}^{(l)}\boldsymbol{h}_v^{(l-1)})$$

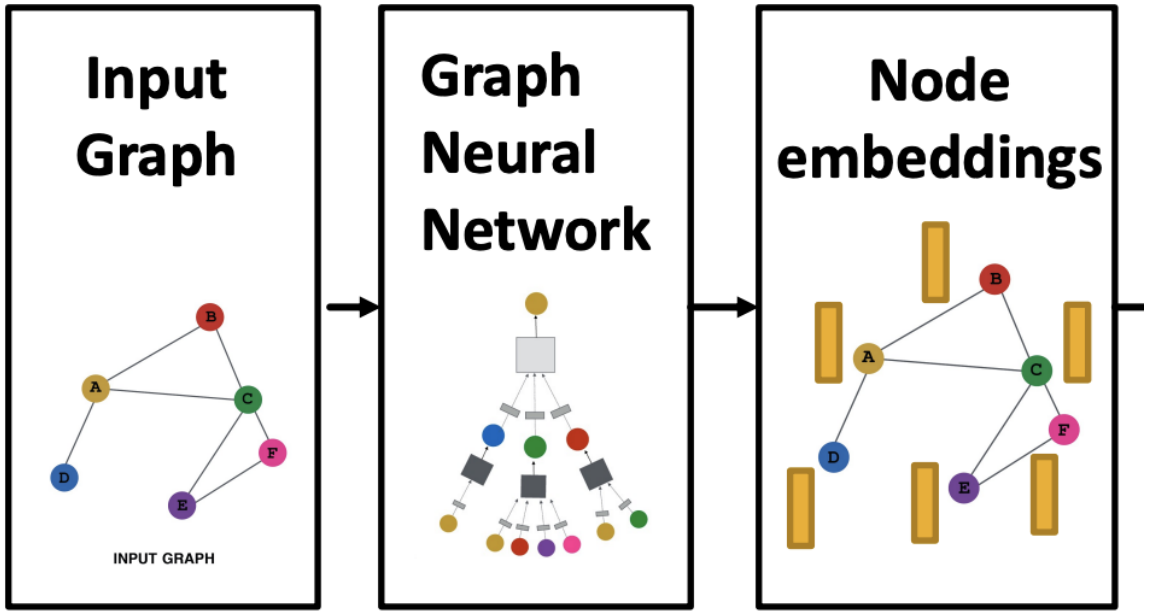
$$\alpha_{vu} = \frac{\exp(e_{vu})}{\sum_{k \in N(v)} \exp(e_{vk})}$$

$$\mathbf{h}_v^{(l)} = \sigma(\sum_{u \in N(v)} \alpha_{vu} \mathbf{W}^{(l)}\mathbf{h}_u^{(l-1)})$$

Training a GNN to perform tasks from different level

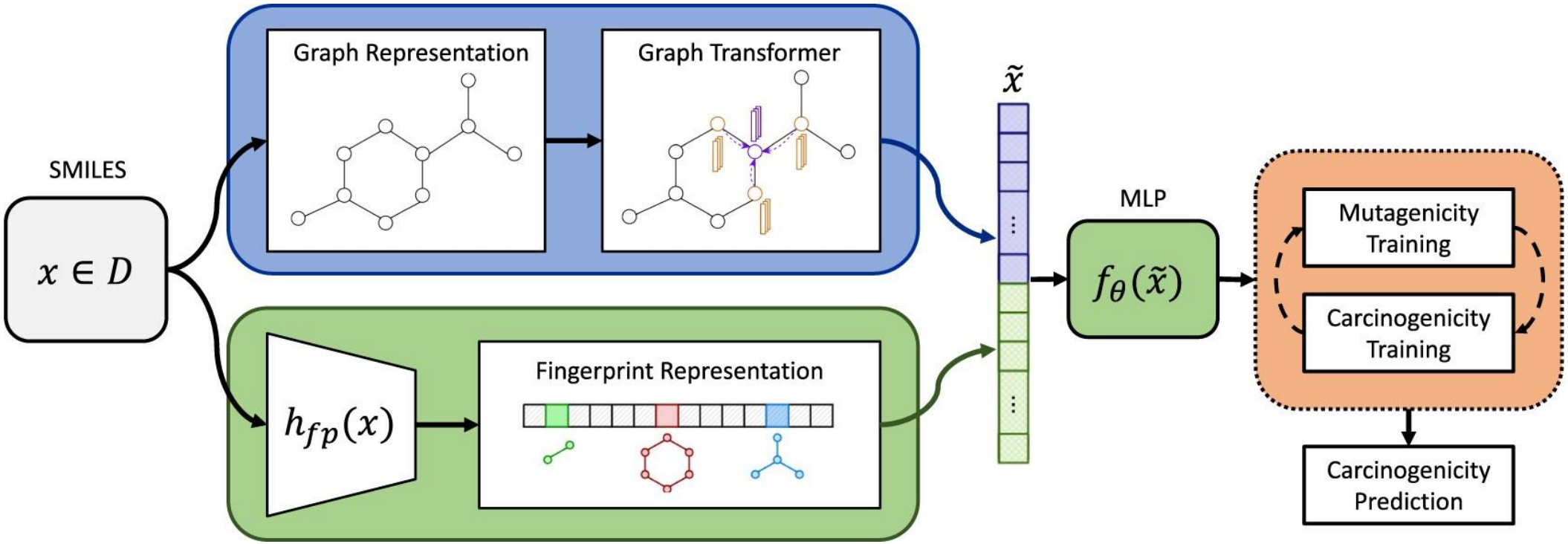
- After convolution, each node will have a specific embedding from the aggregation of information. The embedding of nodes can then be applied to different tasks:

1. Node-level tasks (classify a node using its embedding)
2. Edge-level tasks (predict new edges between nodes using the embedding of the two nodes)
3. Graph-level tasks (classify the whole graph using merged embedding of all nodes)



Application of GNN in computational biology

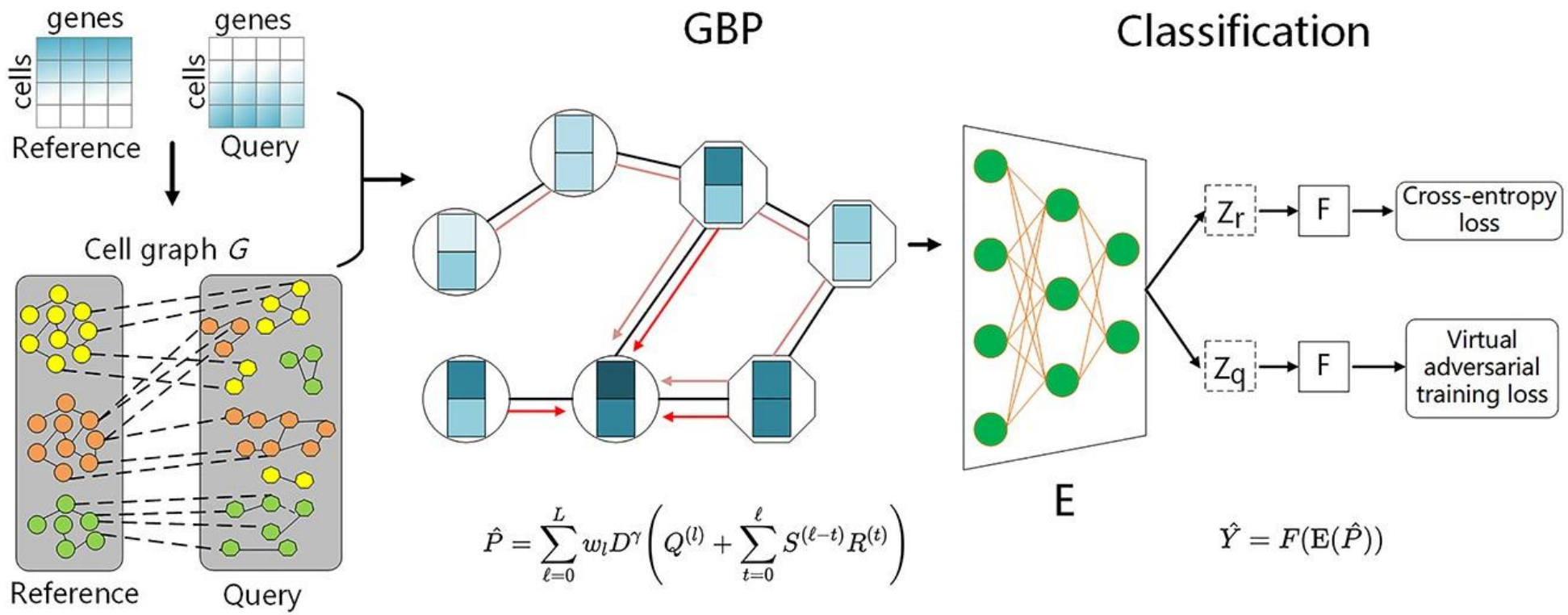
- Molecule carcinogenicity prediction: CONCERTO



Fradkin et al., **Bioinformatics**. 2022

Application of GNN in computational biology

- scRNA-seq cell classification: GraphCS



Zeng et al., **Brief Bioinform.** 2023

Take home message of today

- RNN is capable of reading a sequence input with variable length and generating predictions or new sequences based on context.
- Attention helps alleviate the long-term dependency problem by allowing models to choose which previous context to focus on.
- Transformer further expands the application of attention and uses multi-head self-attention to create an embedding of the input sequence.
- GNN can perform convolution on a graph structure and generate embeddings for all nodes, useful for various levels of tasks.

Suggested reading

RNN:

- https://arxiv.org/abs/1808.03314 (short review on RNN and LSTM)

- Shen Z, Bao W, Huang DS. Recurrent Neural Network for Predicting Transcription Factor Binding Sites. Sci Rep. 2018 Oct 15;8(1):15270. doi: 10.1038/s41598-018-33321-1. PMID: 30323198; PMCID: PMC6189047.

Suggested reading

Transformer:

- https://arxiv.org/abs/1706.03762 (Attention is all you need)

- Two YouTube videos very helpful for understanding the transformer: (strongly recommend 3Blue1Brown)
  - https://www.youtube.com/watch?v=wjZofJX0v4M
  - https://www.youtube.com/watch?v=eMlx5fFNoYc

- Szałata A, Hrovatin K, Becker S, Tejada-Lapuerta A, Cui H, Wang B, Theis FJ. Transformers in single-cell omics: a review and new perspectives. Nat Methods. 2024 Aug;21(8):1430-1443. doi: 10.1038/s41592-024-02353-z. Epub 2024 Aug 9. PMID: 39122952.

Suggested reading

GNN:

- https://github.com/thunlp/GNNPapers?tab=readme-ov-file#chemistry-and-biology (too many papers, select what interests you)

- Fradkin P, Young A, Atanackovic L, Frey B, Lee LJ, Wang B. A graph neural network approach for molecule carcinogenicity prediction. Bioinformatics. 2022 Jun 24;38(Suppl 1):i84-i91. doi: 10.1093/bioinformatics/btac266. PMID: 35758812; PMCID: PMC9235510.

- Zeng Y, Wei Z, Pan Z, Lu Y, Yang Y. A robust and scalable graph neural network for accurate single-cell classification. Brief Bioinform. 2022 Mar 10;23(2):bbab570. doi: 10.1093/bib/bbab570. PMID: 35018408.

Thanks for your listening!