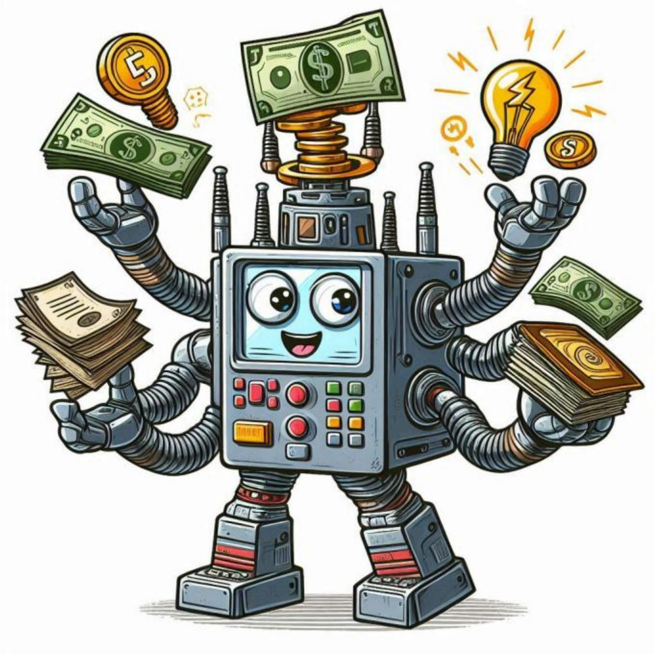
RIVACON

Unboxing Transformers

Forecasting Power Without the Fairy Dust

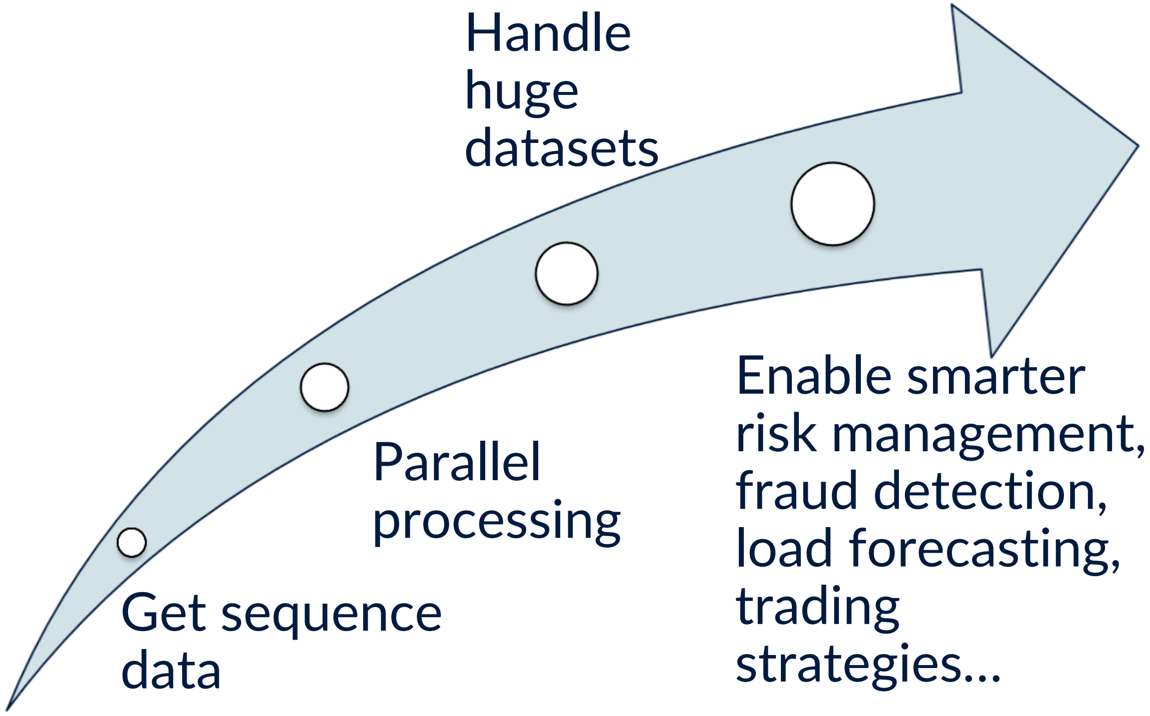


Dr. Fabienne Schmid

August 2025

Image generated by AI

From Fairy Dust to Forecasts: Transformers at Work



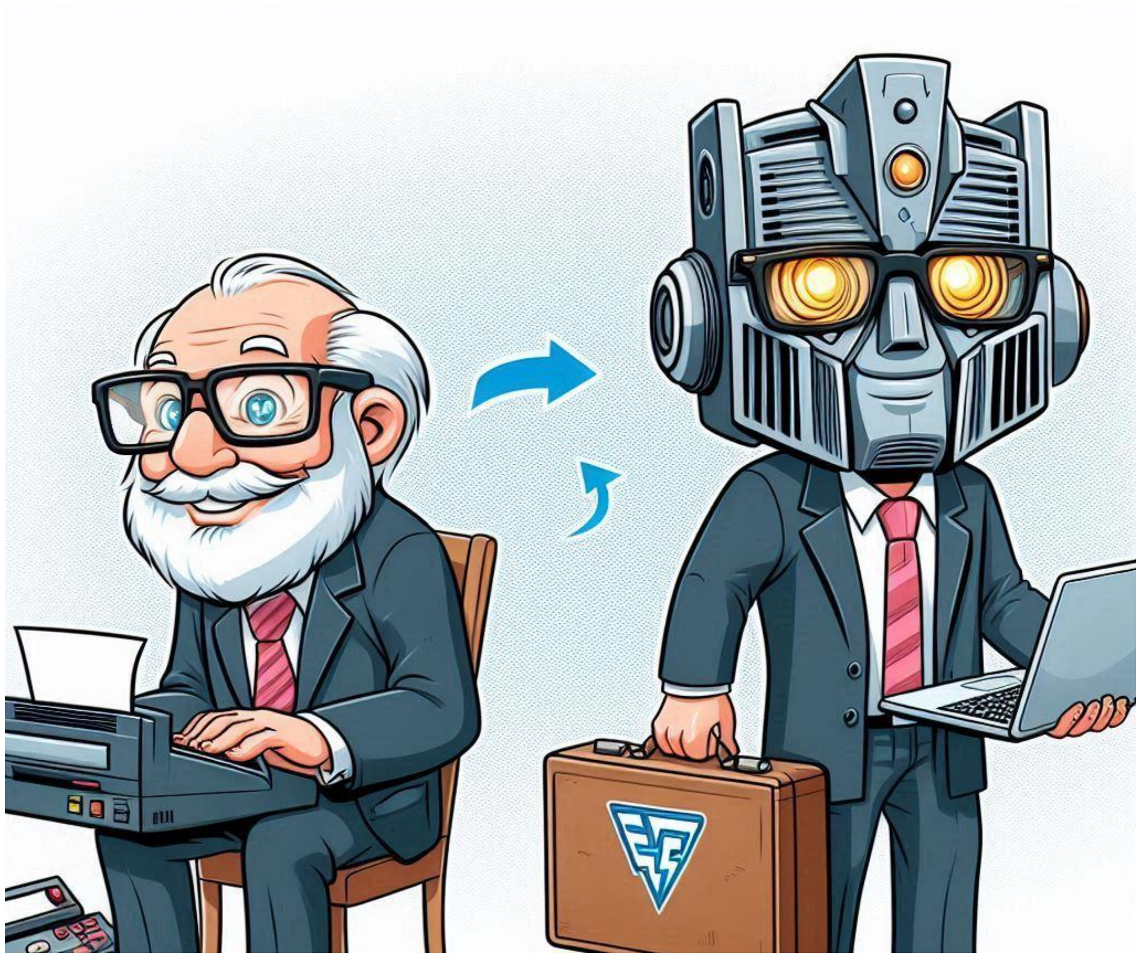


Image generated by AI

The Complexity of Transformers

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

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. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

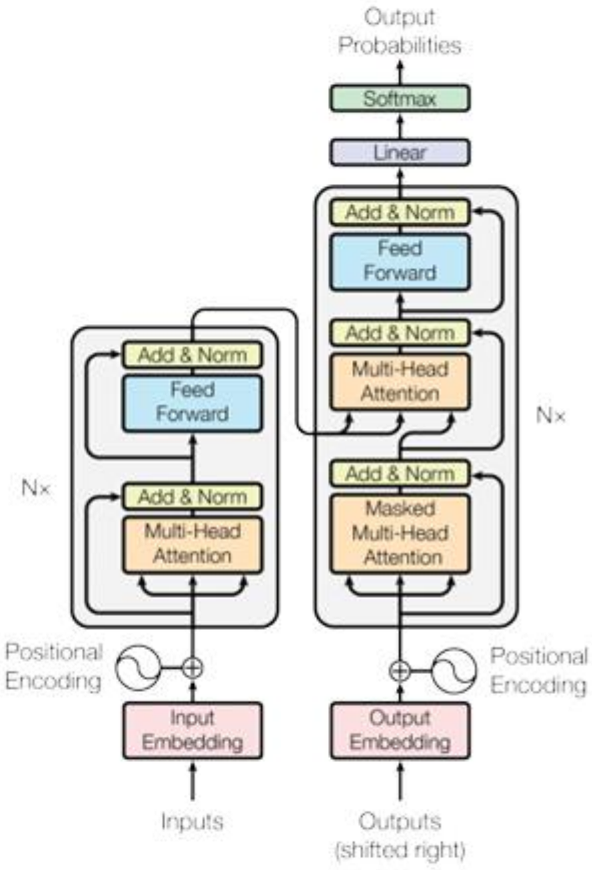
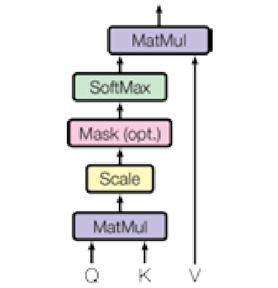
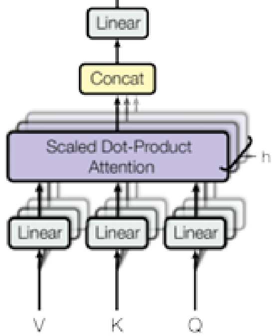


Figure 1: The Transformer - model architecture.





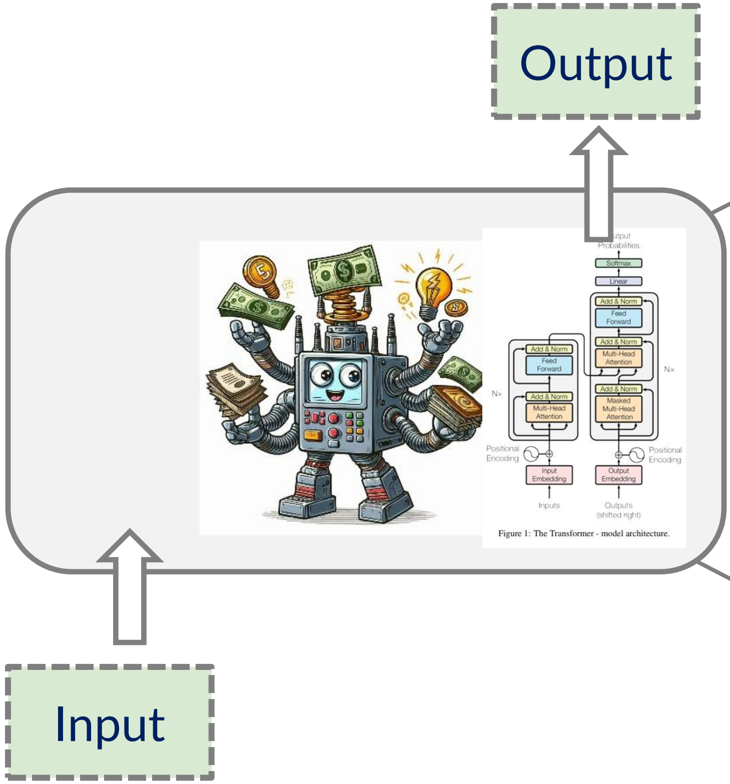
$$\mathrm{Attention}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

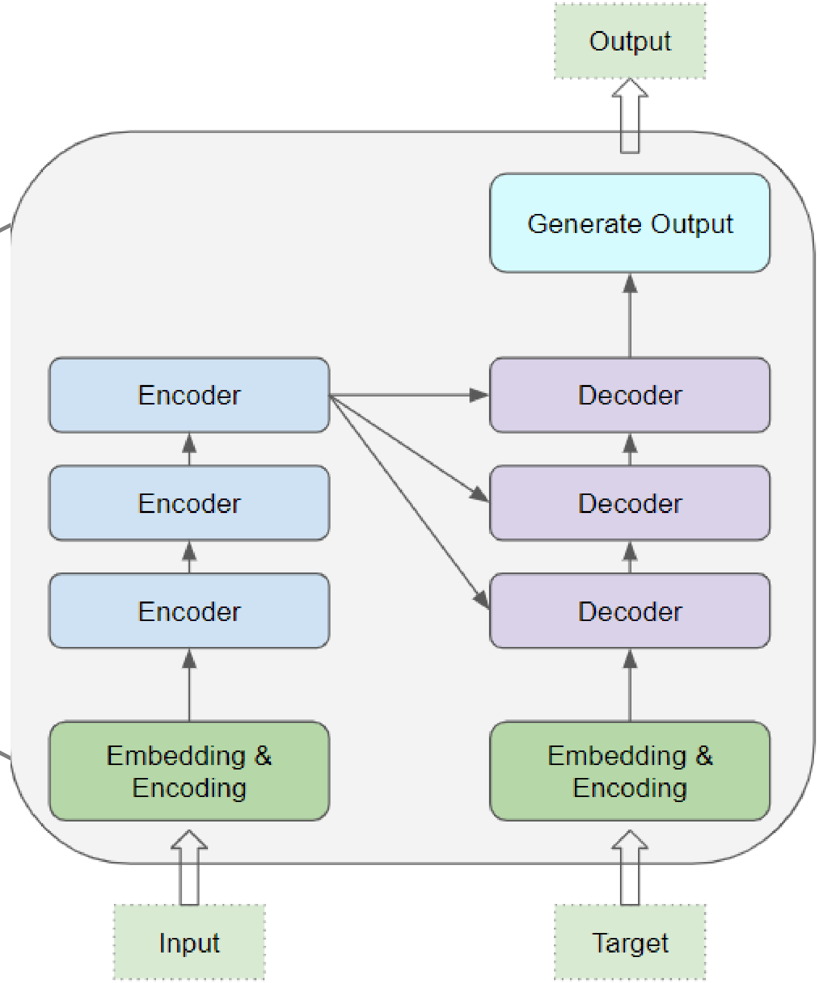
$$\mathrm{MultiHead}(Q,K,V) = \mathrm{Concat}(\mathrm{head_1}, ..., \mathrm{head_h})W^O$$
$$\text{where } \mathrm{head_i} = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

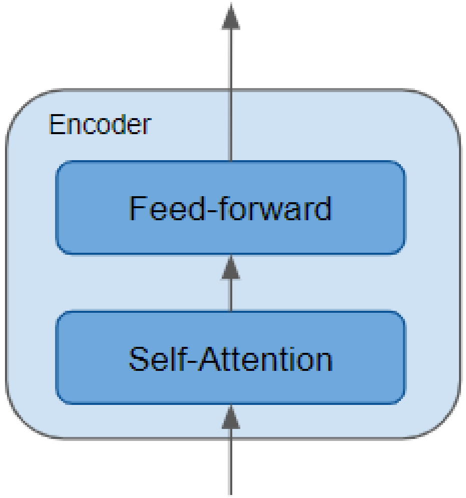
Source: arXiv:1706.03762

Architectural Deep Dive





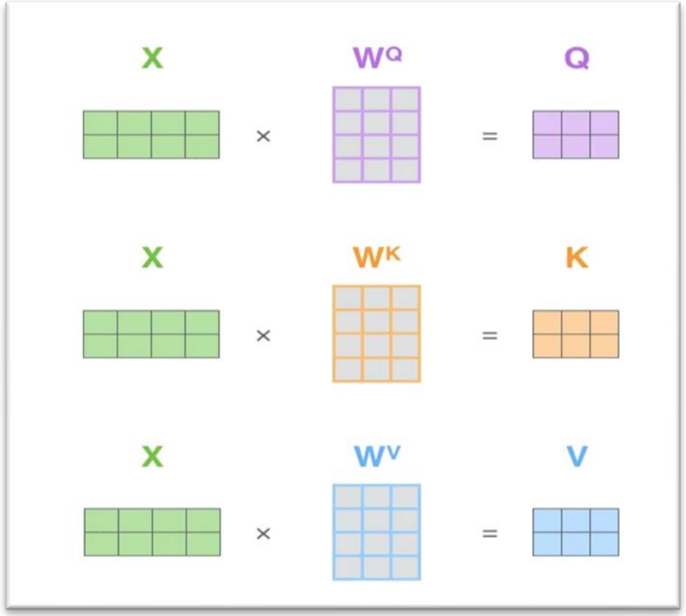
e.g., Encoder:



Source: https://towardsdatascience.com/transformers-explained-visually-part-1-overview-of-functionality-95a6dd460452/

The Attention Mechanism: The Transformer's Secret Sauce

- Attention = focus mechanism
- Lets model highlight relevant parts of input sequence
- Similar to human reading: context matters, not words in isolation:
  - **„The risk manager rejected the *position* because it was too big.“**



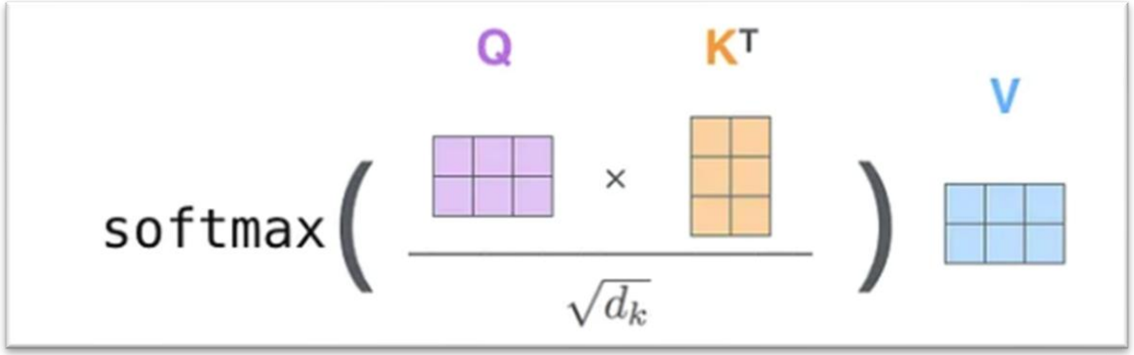
Attention – Think of it as a Library

- Query (Q): Your specific question or research topic
- Keys (K): Keywords or tags on the book spines
- Values (V): The actual content inside the books

Source: https://medium.com/@nitinmittapally/understanding-attention-in-transformers-a-visual-guide-df416bfe495a

The Attention Mechanism: The Transformer’s Secret Sauce

Attention(Q,K,V) =



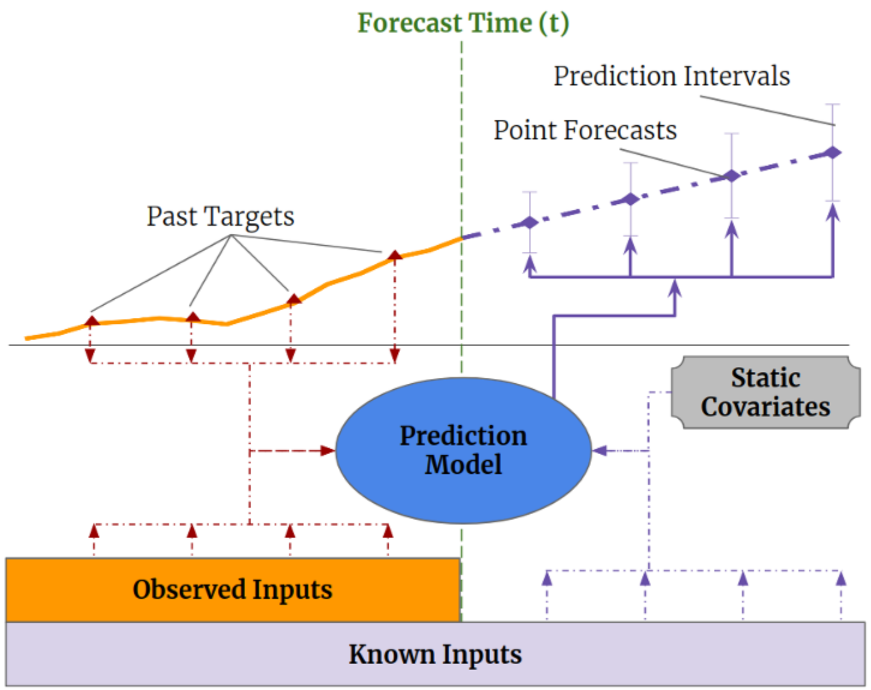
- **Q x K^T** computes the similarity between your query and all available keys *(how relevant each book is)*
- d_k is just a scaling factor to keep the numbers manageable
- **softmax** converts these similarities into percentages *(80% from this book, 15% from that one, 5% from another)*
- **V** contains the actual information you extract (weighted by those percentages)

Source: https://medium.com/@nitinmittapally/understanding-attention-in-transformers-a-visual-guide-df416bfe495a

Transformers in Practice: PyTorch Forecasting

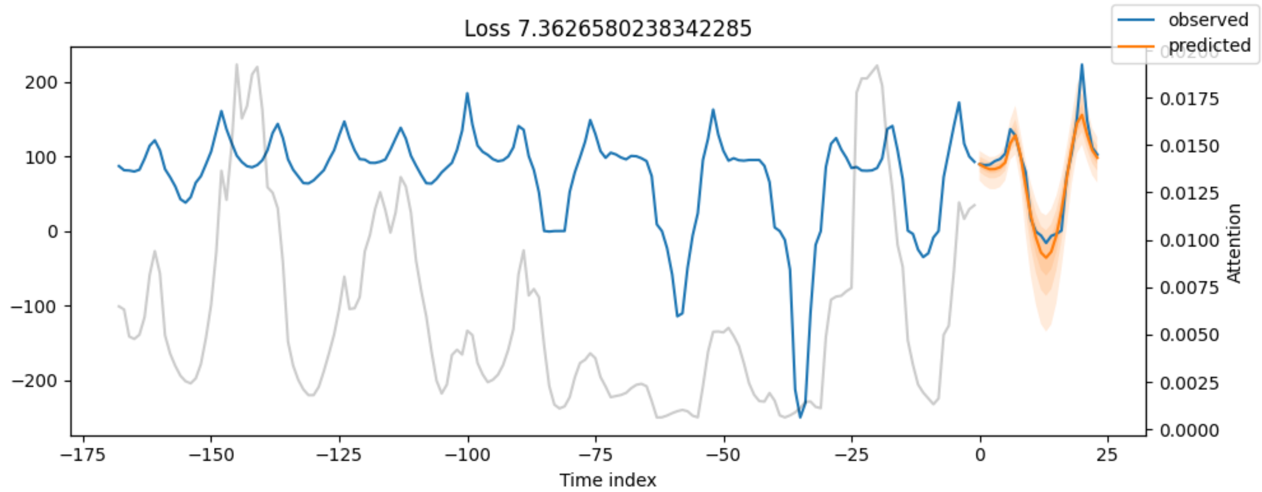
PyTorch Forecasting

- Implementation of Temporal Fusion Transformer (Lim et al., 2020)

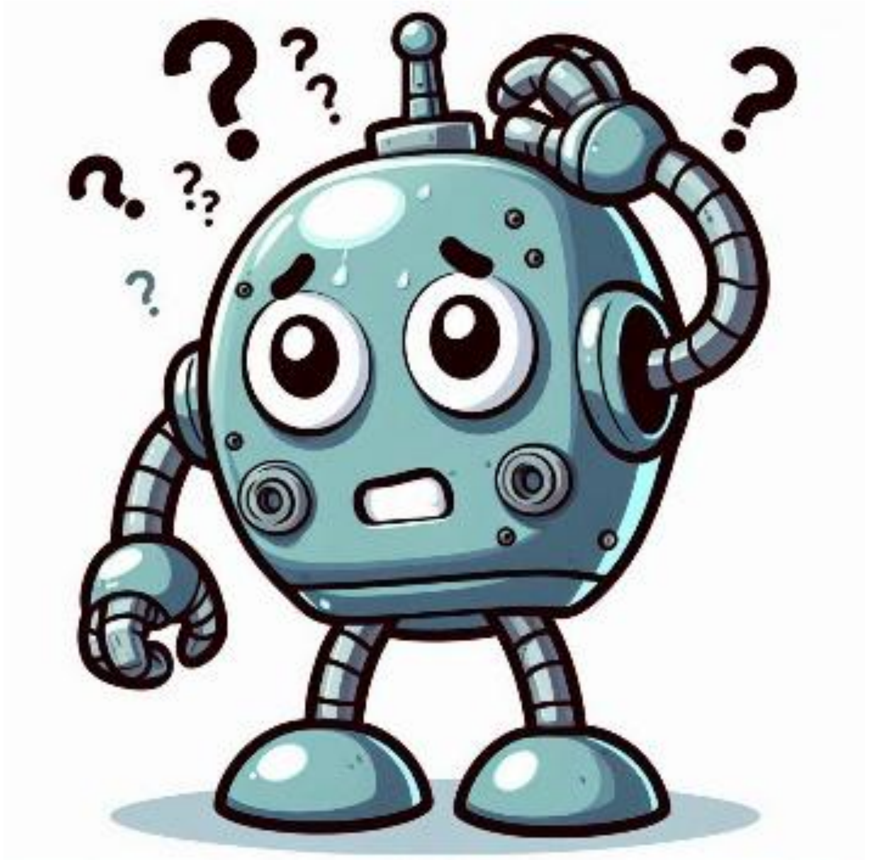


Source: arXiv:1912.09363

Transformers in Practice: PyTorch Forecasting



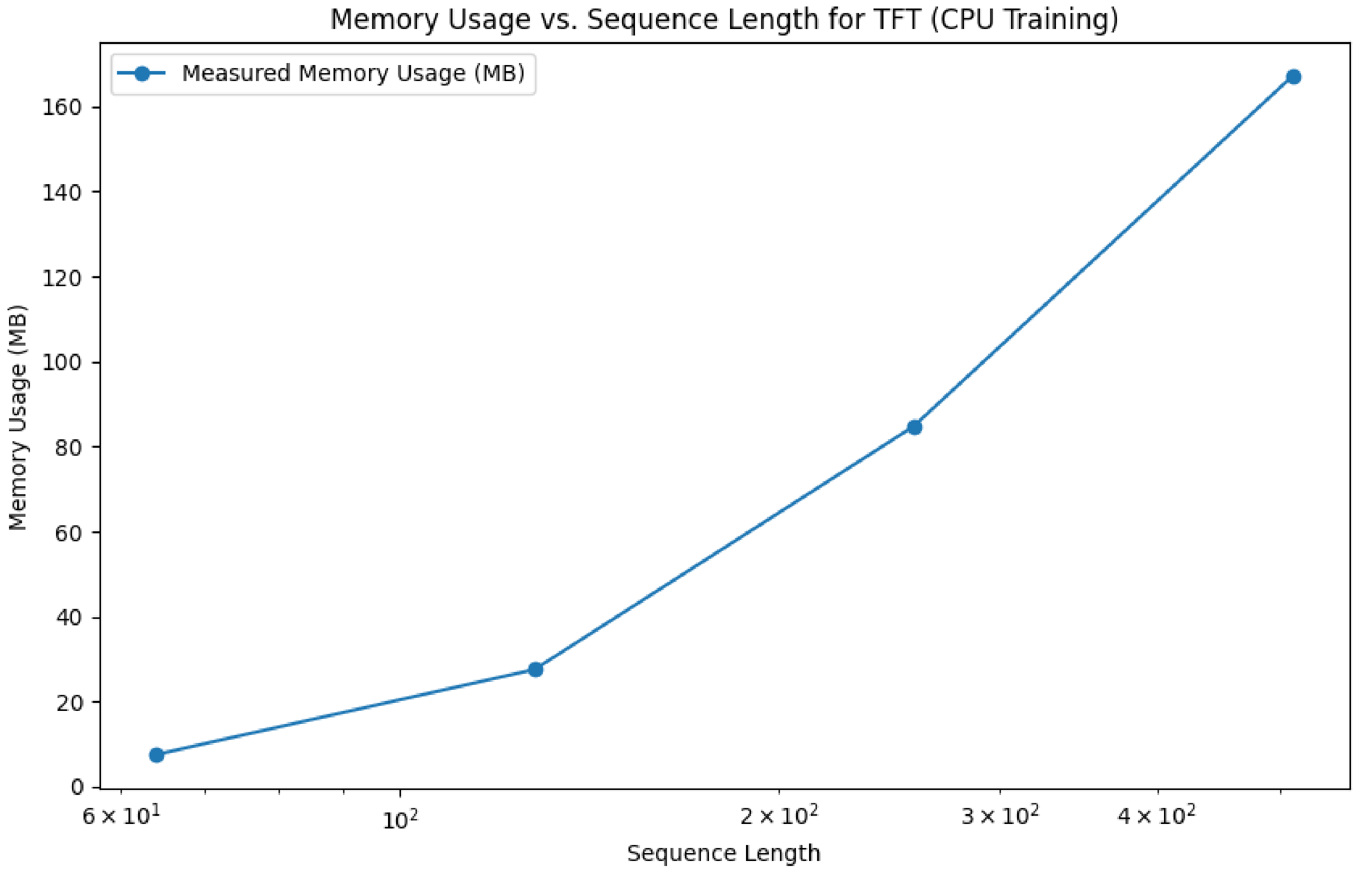
- Forecasts the behavior of the time series, **BUT:** solid theory is missing, i.e., much of it runs on heuristics
- Interpretable Attention: How impactful were past events on today's forecast?



Source: arXiv:1912.09363

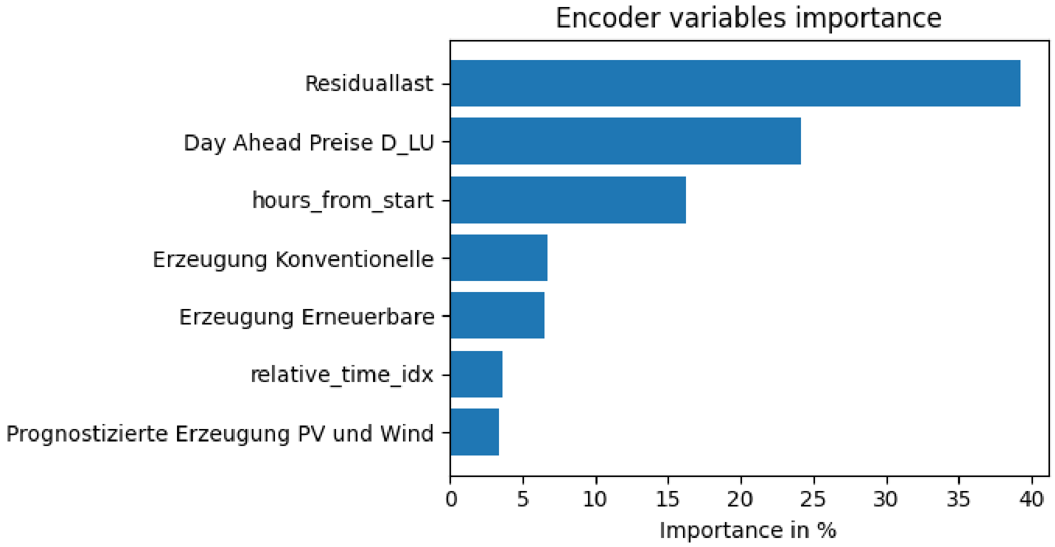
High Memory Demand During Training

- High number of parameters
- Limited context window: strictly limits the available context and prevents memory overflow
- **BUT:** significantly restricts the model's ability to learn from long histories
- Computationally expensive and high energy consumption
- Sustainability and accessibility remain major challenges

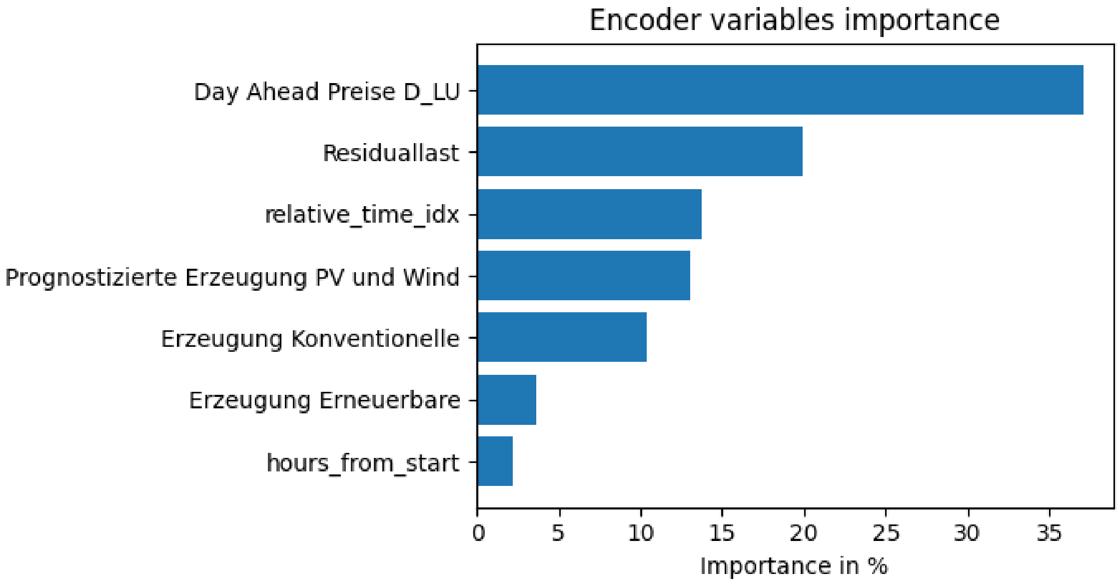


Black-Box Character & Interpretability

```
hidden_size=64,attention_head_size=4,
dropout=0.
```
➢ 432 K Total params




```
hidden_size=16,attention_head_size=1,
dropout=0.1
```
➢ 30.6 K Total params



Summary – No Fairy Dust Required

Powerful tool, BUT:	Garbage In, Garbage Out
Attention Can Be Distracted	Overfitting to Noise
Computationally Expensive	Not Magic

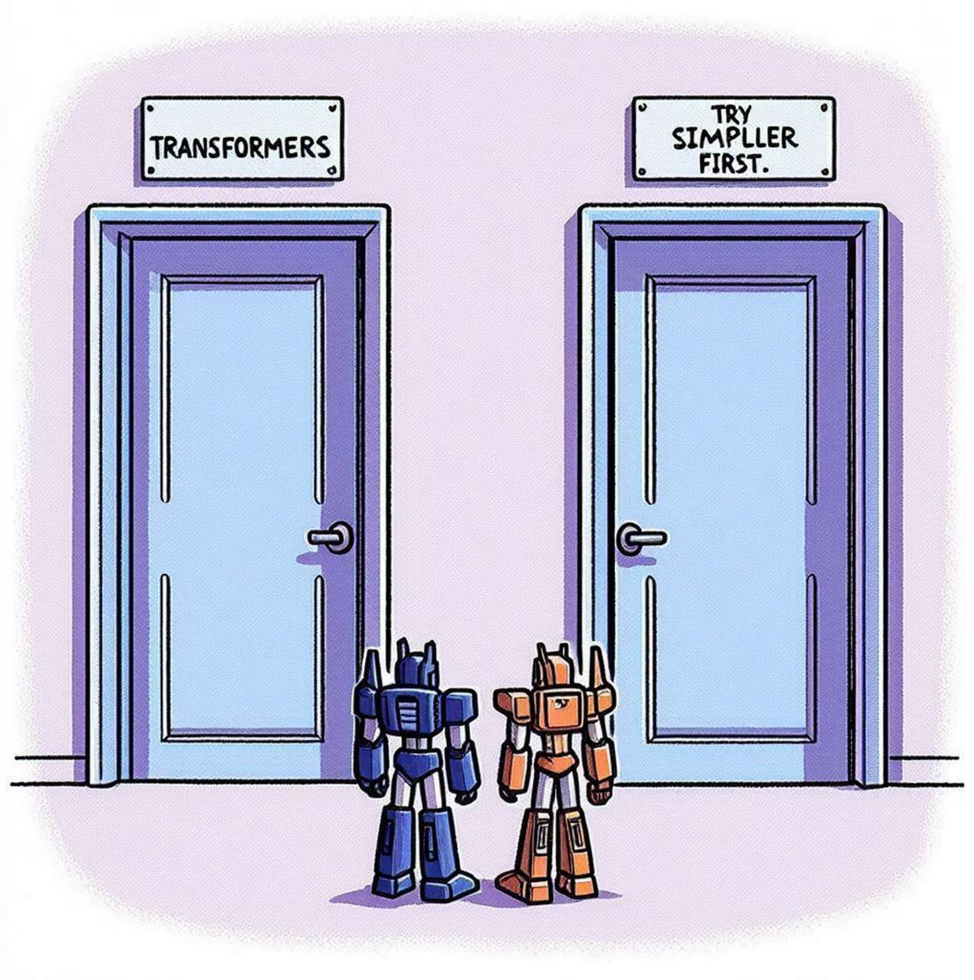


Image generated by AI