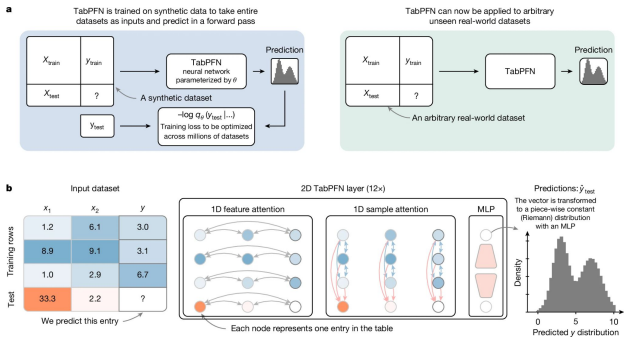
ML module #5 (TabPFN)

Benjamin Sanchez-Lengeling

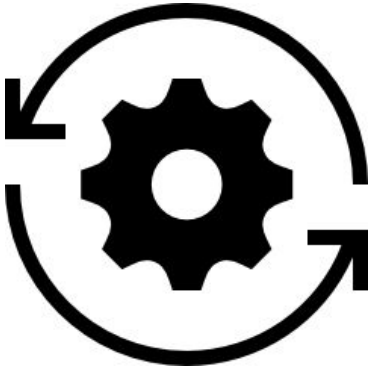
CrossTALK: Cross-Training in AI and Laboratory Knowledge for Drug Discovery.

Panorama



Self-Attention

Pre-training



TabPFN

Working sesh

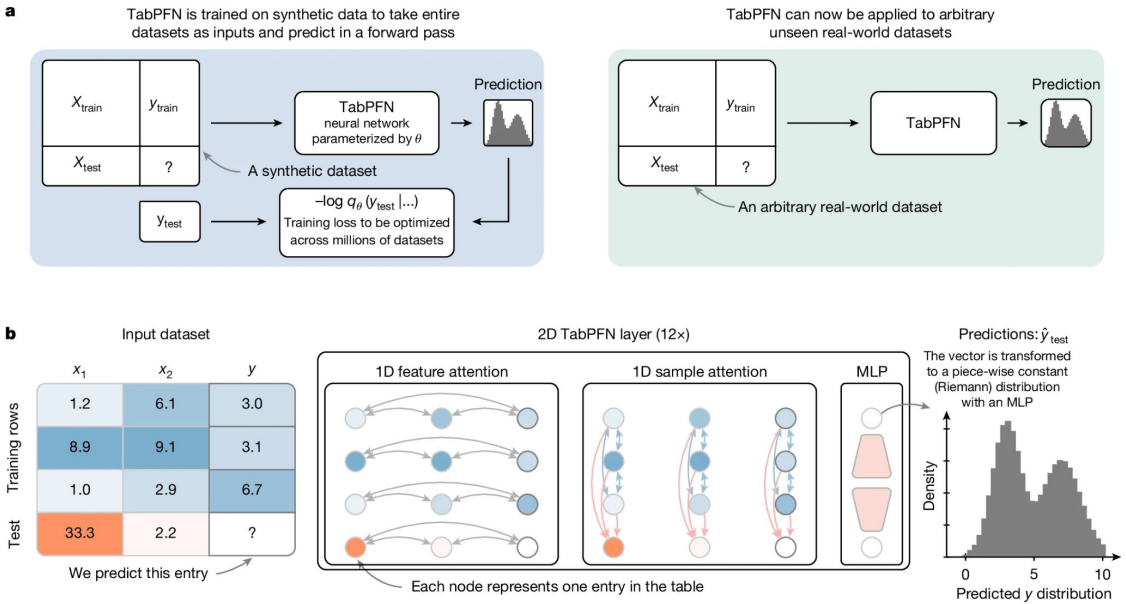
Activity: Talk to someone spatially distant

1. What ideas do you have for modelling?

2. How do we pick a metric to evaluate ML models for drug discovery?

3. What tradeoff do we have to think about when proposing candidate molecules?

TabPFN: Foundation Model for Tabular Data



Important to note:

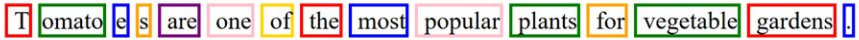
b) Self-attention
a) Pre-training

https://github.com/PriorLabs/TabPFN

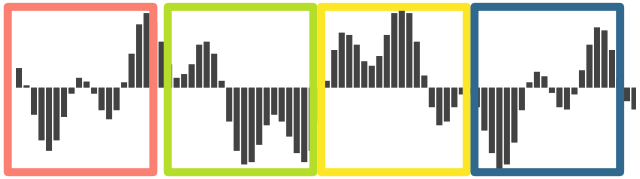
Tokens: Inputs for Transformer/Attention layers

Numerical representations for pieces of data.

Text: Text chunks to embeddings

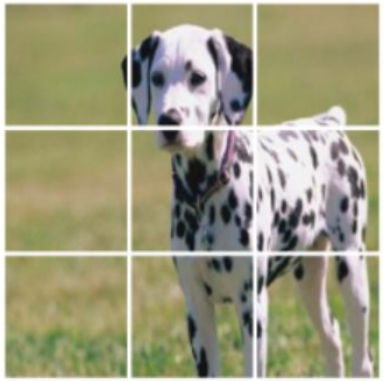


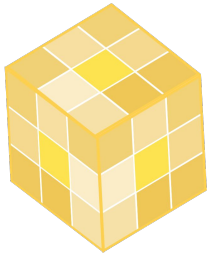
Audio? Sound wave chunks



Graphs? Nodes are tokens

Images: Patches





[Batch, Seq Length, Dim]
3-Tensors

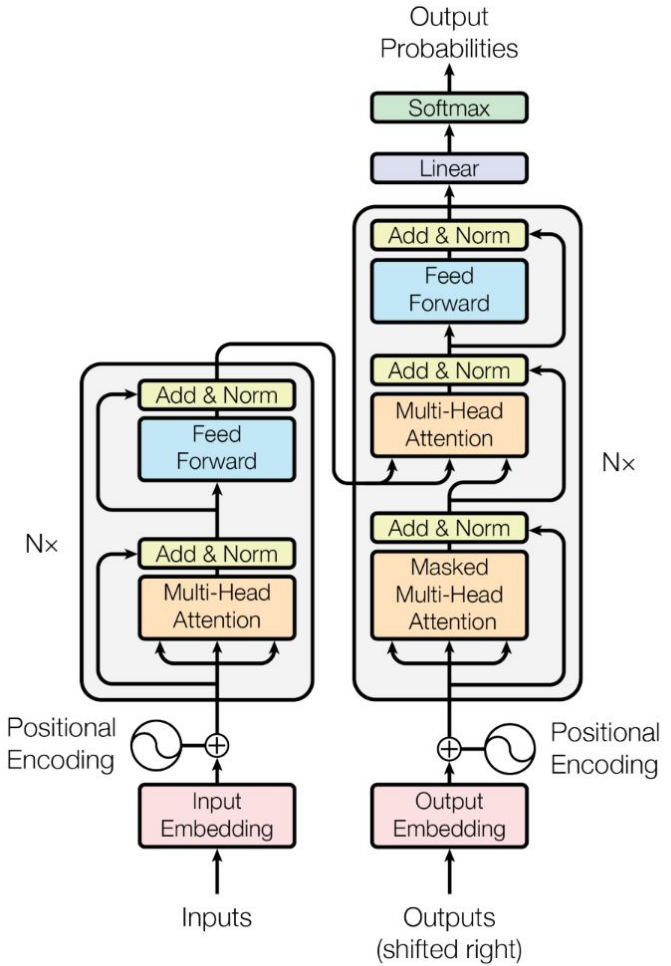
Transformers in a Nutshell

Layers of:

- Attention Mechanism (Multi-headed)
- Positional Encodings

With massive unsupervised datasets:

- Masked self-supervised training
- Contrastive training

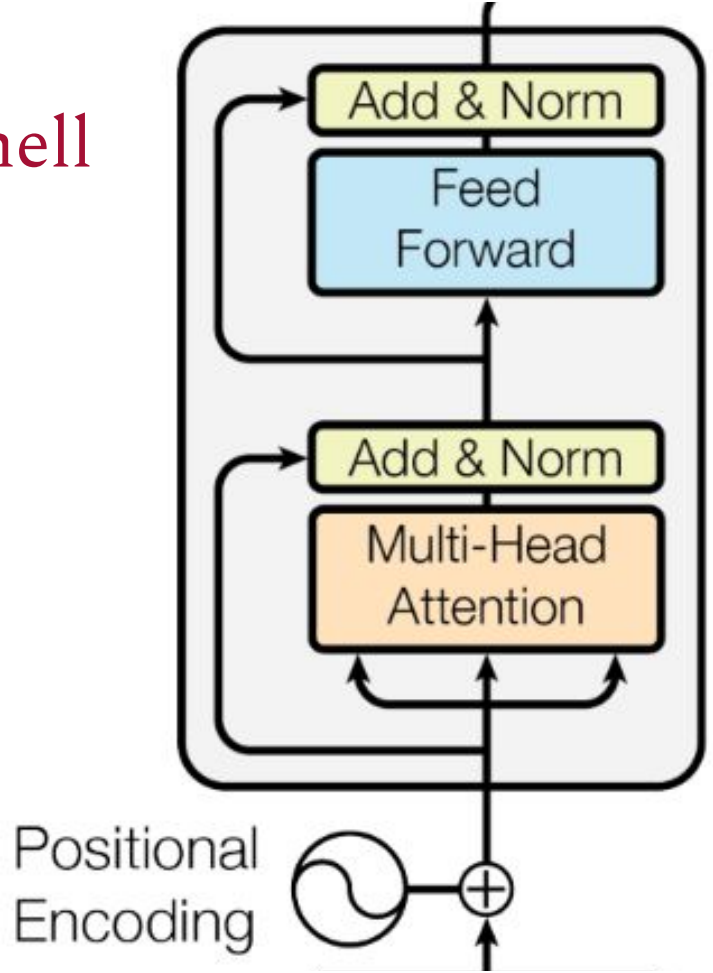


Attention Is All You Need,'17, doi:1706.03762

Transformers Layers in a Nutshell

Inside a transformer layer:

- Self-Attention (multi-headed)
- Residual connections
- Layer Norm
- FFN / MLP
- Positional Encodings



What is Attention? A Bilinear Lens

Attention learns relationships dynamically via “learned” inner products.

$$L(x) = Wx$$

$$B(x, y) = x^T W y$$

$$B(x, x) = x^T W x$$

Bilinear component, n^2 entries

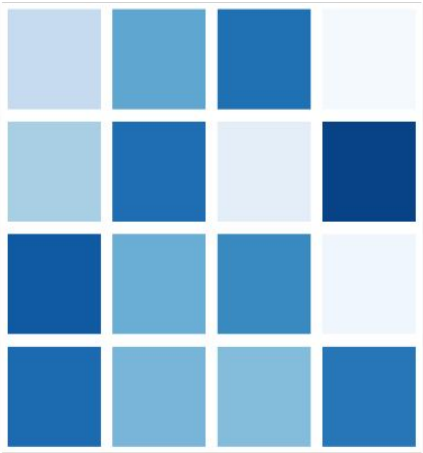
$$\text{score}_{ij} = (q_i \cdot k_j)/\sqrt{d_k}$$

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

Attention Maps: Visualizing Where the Model Attends

Understanding the Focus of Attention Mechanisms

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



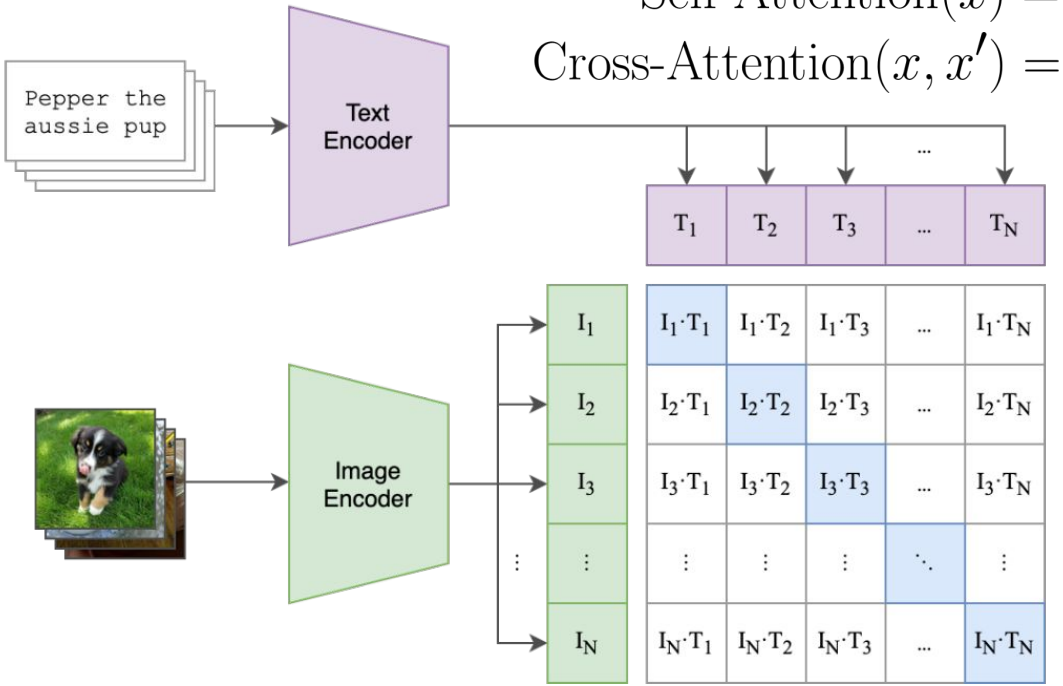
Softmax bias:

- Values between 0 and 1
- Categorical like.
- Attend to one token at a time.

When is Self-Attention not Self? Cross-attention

Applying Attention Between Distinct Inputs

$$\text{Self-Attention}(x) = \mathit{Attention}(Q(x), K(x), V(x))$$
$$\text{Cross-Attention}(x, x') = \mathit{Attention}(Q(x), K(x'), V(x))$$



- Queries can be one set of tokens
- Keys can be another set of tokens
- Attention scores might not be square
- Values depend on the output
- Same equations apply

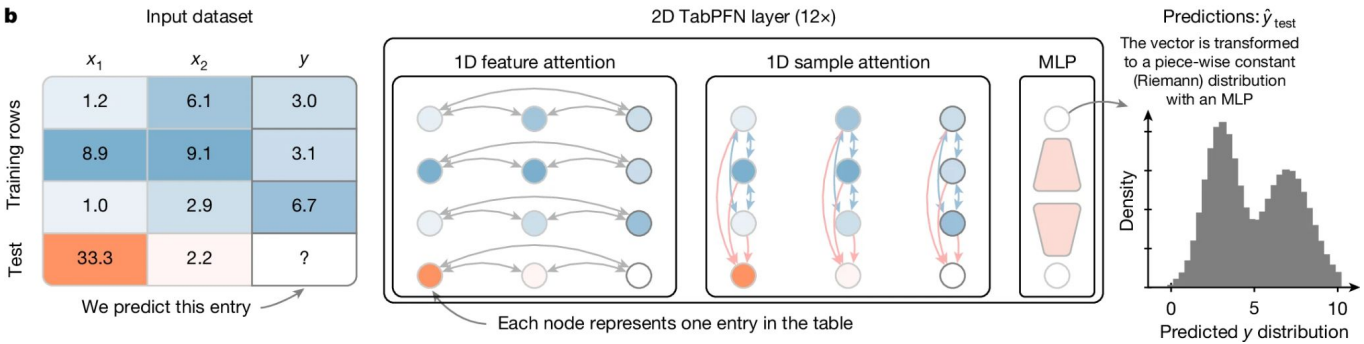
Learning Transferable Visual Models From Natural Language Supervision, ‘21, doi:2103.00020

Self-Attention recap

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

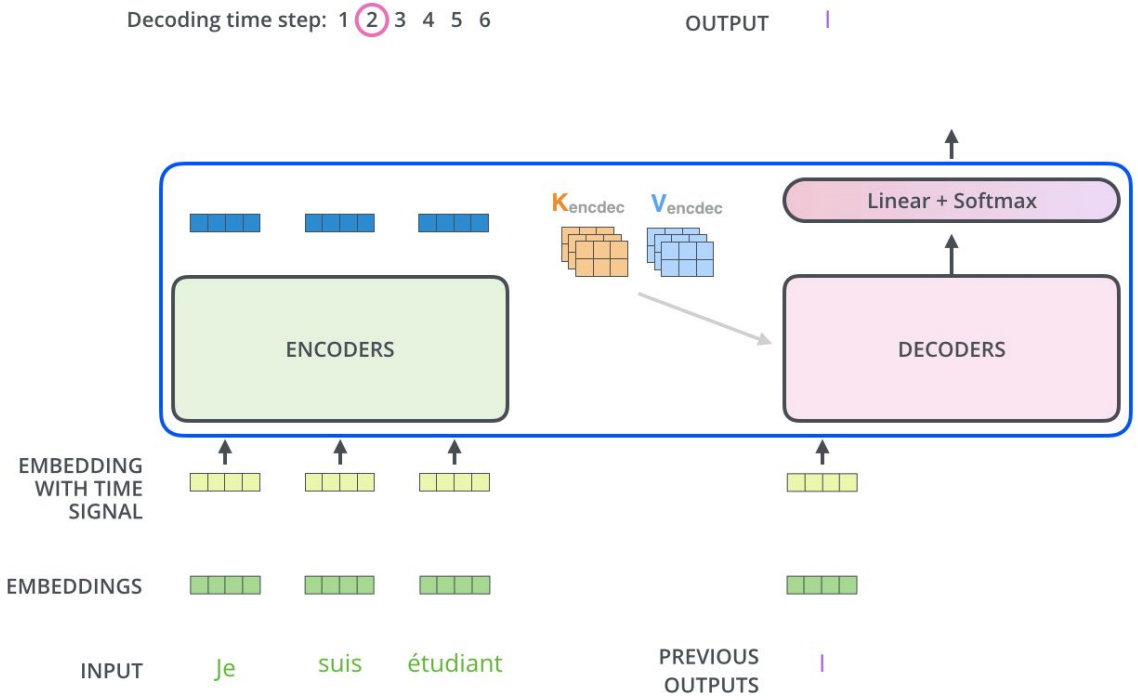
$$\text{Self-Attention}(x, mask) = \text{Attention}(Linear(x), Linear(x), Linear(x), mask)$$

$$\text{Cross-Attention}(x, x', mask) = \text{Attention}(Linear(x), Linear(x'), Linear(x), mask)$$



Outputs: Auto-Regressive Decoding of Tokens

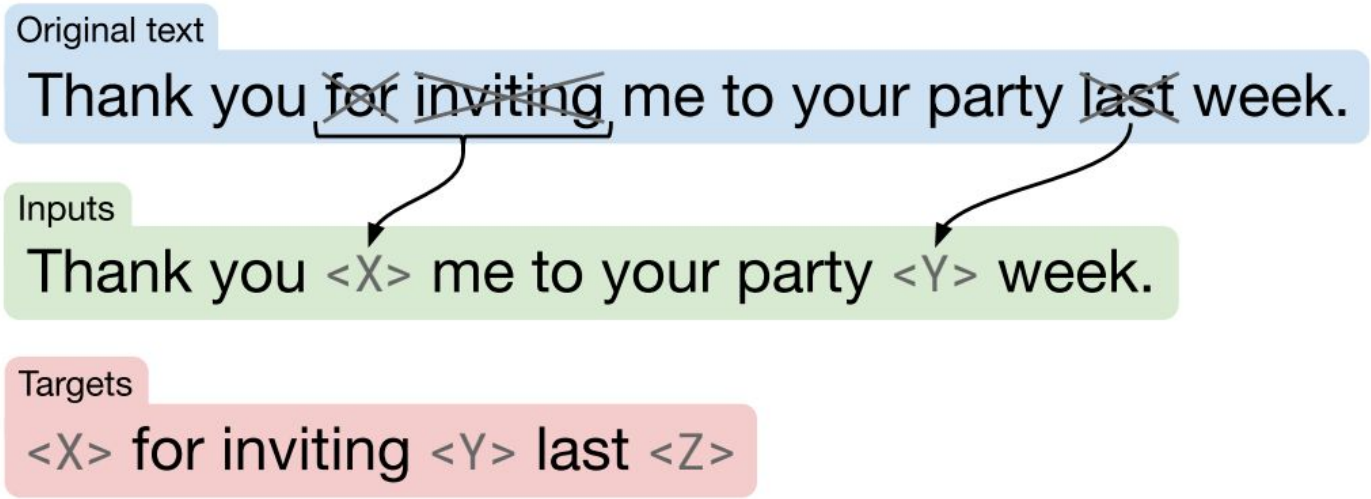
Decoding one token at a time, using previous outputs.



https://jalammar.github.io/illustrated-transformer/

Pre-training: Masked Language Modelling (Self-supervised)

Span denoising in T5x as an example

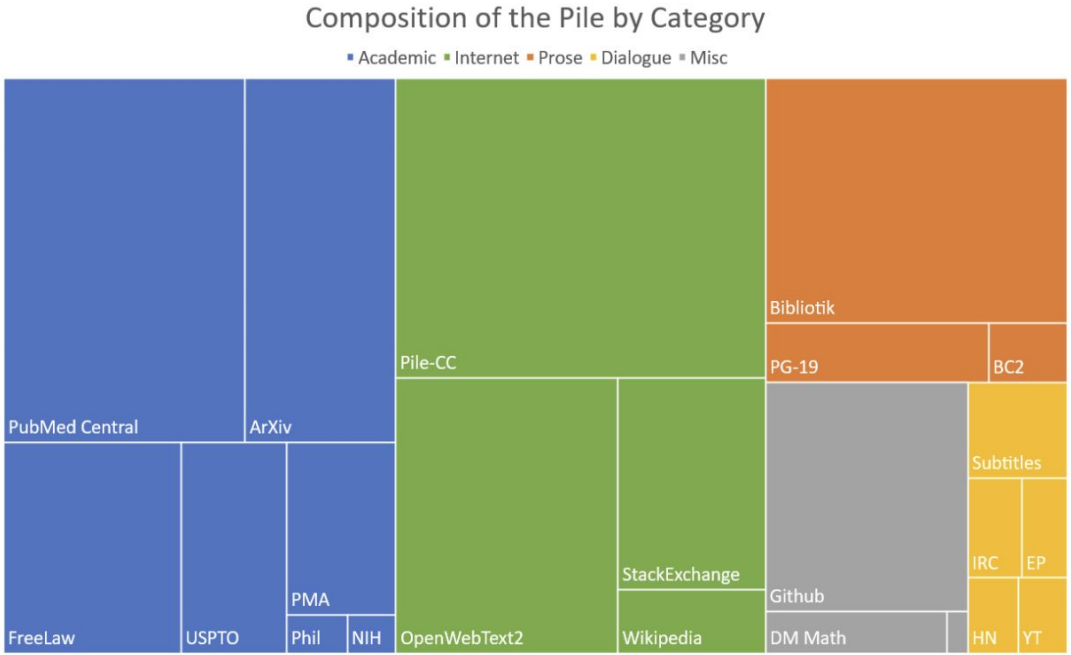


Scaling Up Models and Data with t5x and seqio, ‘22, arXiv:2203.17189

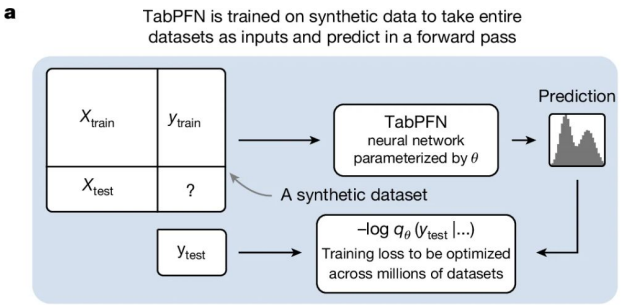
Sizes of Text Datasets for LLMs

Scale significantly impacts model ability.

Common Crawl	225 B web pages	275 TBs
The Pile	22 datasets	825 GiB



TabPFN's Pre-Training: From Solving One Problem to Learning How to Solve *Any* Problem

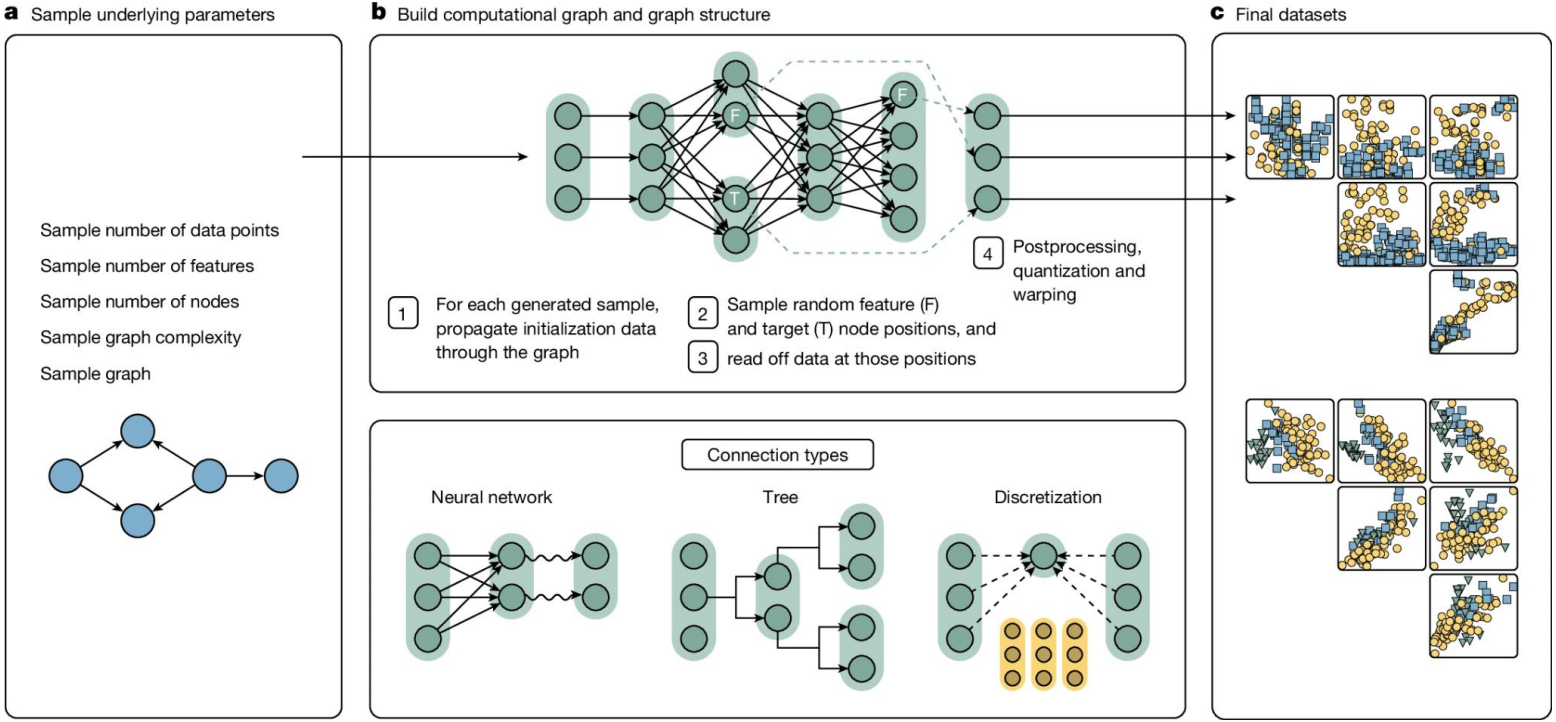


A **single, powerful model** is pre-trained once on millions of *simulated* datasets.
It learns a universal algorithm for finding patterns in tabular data.

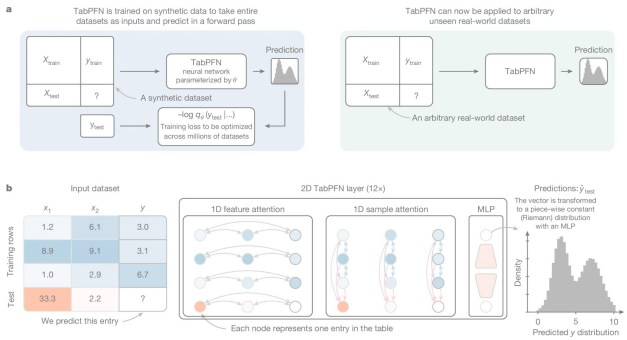
The core mechanism is **in-context learning**. At inference time, TabPFN doesn't update its weights; it uses your training data as the instruction manual for how to make the next prediction.

- For a Large Language Model (LLM), the **"Prompt"** is a string of text.
- For TabPFN, the **"Prompt"** is your entire training dataset (X_train, y_train).

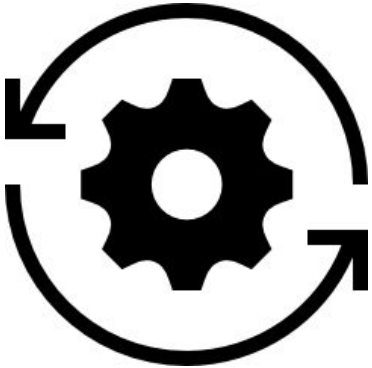
Synthetic dataset generation via Structured Causal Models



Panorama



TabPFN



Working sesh

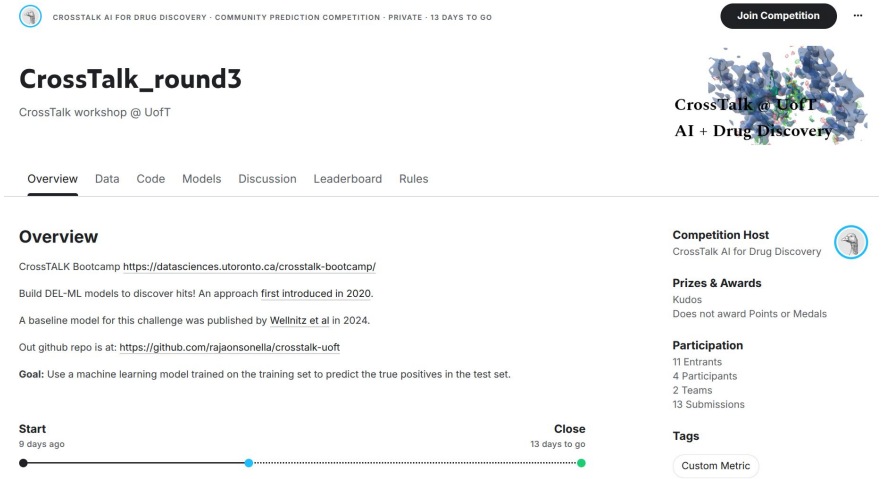
Results...
Drumrolls

Leaderboard

Old metric

1	Oleksii Nakhod		0.93482	18	13h
2	Walter Virany		0.91056	4	4d
3	Seshill Real Armas		-44.00000	2	10h
4	UofTShengqing		-47.00000	6	11h
5	Sherrrrr123		-47.00000	1	1h
6	Rach999		-48.00000	1	1h
7	Shilpa Yadahalli		-50.00000	1	4d

Kaggle: Updating the penalty later today



“Penalized”
Hits @ 200

For color palette

https://colorbrewer2.org/#type=sequential&scheme=BuGn&n=3

Dark2

Set3

Cividis (continuous)

PiYG (divergent)