Foundation Models and ICL

Lecture 13 - 36th International Summer School SAA
University of Lausanne

Ronald Richman,
Salvatore Scognamiglio, Mario V. Wüthrich

Friday, 12 September 2025

What Are Foundation Models?

- Definition: Models trained on broad, diverse data (often self-supervised) that transfer across many downstream tasks with minimal task-specific changes
- Core properties: scale (parameters, tokens), general representations, and versatile adaptation (prompting/ICL, PEFT, RAG)
- Backbone: For NLP, usually `Decoder` Transformers are used with AR generation (Vaswani et al., 2017)
- Scaling: Capability follows compute/data scaling laws (Kaplan, McCandlish, Henighan, Brown, et al., 2020); compute-optimal training balances model and token budgets (Hoffmann, Borgeaud, Mensch, et al., 2022)

Architectural Backbone: Transformers

- Self-attention replaces recurrence/convolution; parallelizable sequence modeling (Vaswani et al., 2017)
- Pretraining variants: masked LM (Devlin, Chang, Lee, & Toutanova, 2019) vs. autoregressive decoders
- Inductive biases: attention as data-dependent mixing; KV-memory view (Geva et al., 2021)
- Positional encodings and other long-context tricks (e.g., RoPE; Su, Lu, Pan, Wen, & Liu, 2021)
- Previously we have mentioned scaling laws (Kaplan et al., 2020)

Data Needs for Training FMs

- Scale: billions to trillions of tokens with a compute-optimal balance of parameters and tokens; budget tokens, not only params (Hoffmann et al., 2022).
- Diversity: broad domain, style, language, and modality coverage to learn general representations and handle distribution shift.
- Quality: rigorous filtering and deduplication at document and paragraph level; near-duplicate removal; language ID; low-quality and boilerplate removal; test-set decontamination.
- Long-tail coverage: upweight rare languages, domains, and entities; mix high-quality curated slices with broad web-scale data to raise signal-to-noise.

FMs as a Model Class

- Model class: $\mathcal{F} \coloneqq \{f_\theta : \mathcal{X} \to \mathcal{Y} \mid \theta \in \Theta\}$ with $\Theta \subset \mathbb{R}^P$, $P \gg 10^8$
- Probabilistic view: f_θ parameterizes $p_\theta(y \mid x)$; prediction by $\hat{y} = \mathbb{E}_{p_\theta}[Y \mid x]$ or, more commonly, sampling from the last softmax layer with various approaches
- Self-supervised pretraining defines surrogate labels; the same $\mathcal{F}$ shared across modalities

Pretraining and Adaptation

- Pretraining (AR MLE): $\max_\theta \sum_t \log p_\theta(x_t \mid x_{<t})$ calibrates broad priors
- Adaptation options: fine-tune, PEFT (low-rank), ICL via exemplars, RAG via retrieval
- ICL view: $\hat{y} = f_{\theta^\star_{\text{pre}}}([\mathcal{C}, x])$ with $\mathcal{C}=$ in-prompt demonstrations (Brown et al., 2020)

From GPT-1 to GPT-4

- GPT-1/2: Unsupervised pretraining improves transfer; scaling unlocks fluent generation (Radford, Narasimhan, Salimans, & Sutskever, 2018; Radford et al., 2019)
- GPT-3: Emergent few-shot learning via prompting; broad task coverage without gradient updates (Brown et al., 2020)
- InstructGPT: Alignment via RLHF improves following instructions and safety (Ouyang et al., 2022)
- GPT-4: Strong reasoning, broader safety/robustness; multimodal variants (OpenAI, 2023)

GPT-3: Few-Shot Learners

- Scale: 175B parameters; diverse pretraining corpus (Brown et al., 2020)
- Modes: zero-shot, one-shot, few-shot; strong performance without finetuning
- Sensitivities: prompt format/order; benefits from better instructions and demonstrations
- Limitations: calibration and factuality; improved downstream via RAG and alignment

InstructGPT: RLHF Alignment

- Pipeline: supervised fine-tuning (SFT) $\rightarrow$ reward model $\rightarrow$ RL (PPO) (Ouyang et al., 2022)
- Effect: better instruction following, reduced toxicity; modest performance trade-offs mitigated by scale
- Practice: preference data quality and coverage critical; monitor reward hacking

Few-Shot and Zero-Shot with CoT

- Few-shot prompting (GPT-3): in-prompt exemplars enable rapid task adaptation (Brown et al., 2020)
- Zero-shot Chain-of-Thought: reasoning cue elicits stepwise solutions (Kojima, Sagawa, Lu, et al., 2022)
- Self-consistency: sample multiple chains and vote to improve accuracy (X. Wang, Wei, Schuurmans, et al., 2022)

Working with FMs - 1

- Prompting and few-shot ICL: task specification at inference
- Parameter-efficient fine-tuning (PEFT): adapters/LoRA conceptually
- Retrieval-augmented pipelines: ground outputs; reduce hallucinations
- Serving: KV-cache optimization, batching, speculative/parallel decoding

Working with FMs - 2

- Instruction prompts with role/content separation; few-shot exemplars for schema priming
- Reasoning cues: CoT and self-consistency for multi-step tasks
- Guardrails: constrained decoding, refusal policies; retrieval for factual grounding
- Evaluation: hold-out tasks, calibration checks, distribution-shift probes

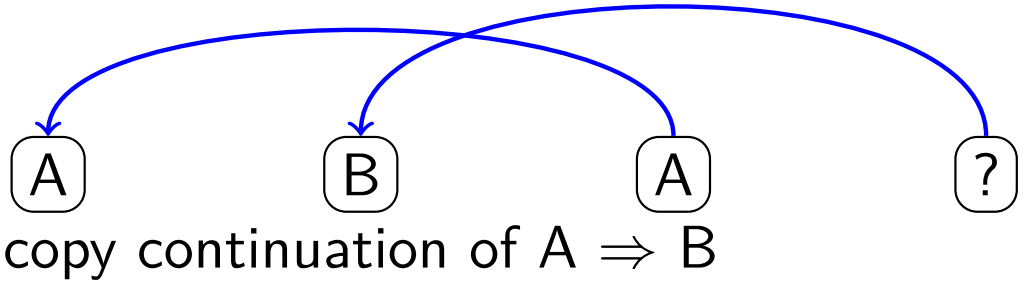
ICL: Emergence and Setup

- Observation (GPT-3): decoder-only LMs adapt to new tasks from in-prompt exemplars without weight updates (Brown et al., 2020)
- Prompt pattern: $[(x_1, y_1), \ldots, (x_K, y_K), x_{\text{query}}] \mapsto \hat{y}_{\text{query}}$
- Capabilities: few-shot classification, translation, QA; sensitivity to exemplar order and format
- Scaling effect: reliability and breadth improve with model/data scale

ICL: Broader Evidence and Limits

- Role of demonstrations (Min, Lewis, Zettlemoyer, & Hajishirzi, 2022): label-space priming, instruction format, and answer options drive gains
- Simple function classes (Garg, Tsipras, Roelofs, Hazan, et al., 2022): Transformers can learn linear/affine rules in-context under suitable pretraining
- Takeaway: ICL performance depends on prompt design, distributional match, and model scale

Induction Heads: Schematic



Induction heads attend to earlier repeated tokens and copy their continuations (Olsson, Elhage, Nanda, et al., 2022).

Why Tabular is Different

- Heterogeneous features, mixed types, missingness, and no natural order
- Not clear how to train across tables!
- Strong baselines (GBDT) set high bar; sample sizes often modest
- Foundation approach: pretrain cross-table priors and reuse across tasks

Tabular Landscape: Families and Design

- TabTransformer: categorical tokenization + attention over features (Huang, Khetan, Cvitkovic, & Karnin, 2020)
- FT-Transformer: added continuous features leading to simplified, strong baseline for tabular DL (Gorishniy, Rubachev, Khrulkov, & Babenko, 2021)
- TransTab: cross-table transfer with aligned embeddings (Z. Wang & Sun, 2022)
- Design: feature masking/denoising, schema-agnostic tokens, missingness augmentation

Tabular foundation models: the landscape

TabPFN. Prior-data fitted network for tabular classification and beyond. Trains on synthetic tasks sampled from a *prior over generative processes*; uses alternating column and row attention to perform ICL on full tables in a single forward pass.(Hollmann, Müller, Eggensperger, & Hutter, 2022; Müller, Hollmann, Pineda Arango, Grabocka, & Hutter, 2021)

TabPFN v2. Nature 2025: tabular foundation model with wide wins up to $n \leq 10{,}000$ and strong calibration, large speedups over classical baselines.(Hollmann et al., 2025)

TabICL. ICML 2025: scalable ICL to n in the 10^4 to 5×10^5 range by a 2-stage architecture: column-then-row embedding to fixed-dimension, then a transformer for ICL. Often faster and stronger than TabPFN for large n.(Qu, Holzmüller, Varoquaux, & Le Morvan, 2025)

TabPFN: amortized Bayesian view

Prior-data fitted network learns to approximate the Bayes posterior predictive under task prior Π:

$$p(y^\star \mid \mathbf{x}^\star, \mathcal{D}) \;=\; \int p(y^\star \mid \mathbf{x}^\star, \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathcal{D})\, d\boldsymbol{\theta}.$$

PFN learns a function $f_\psi(\mathbf{x}^\star, \mathcal{D}) \approx p(y^\star \mid \mathbf{x}^\star, \mathcal{D})$ by minimizing the expected negative log likelihood over synthetic tasks sampled from Π.(Müller et al., 2021)

- *ICL* emerges: the provided table acts as context; no gradient step at inference.
- Prior Π can encode causal structure, class-imbalance, feature types, noise, or temporal drift.(Helli, Schnurr, Hollmann, Müller, & Hutter, 2024)

TabPFN: architecture and objective

Input. A table $\mathbf{X} \in \mathbb{R}^{n \times d}$ and targets $\mathbf{y} \in \mathcal{Y}^n$ (some labels masked for query points).

- Alternating attention over columns and rows to mix feature-wise and record-wise information efficiently.

Training objective. For classification with classes $\{1, \ldots, K\}$:

$$\min_{\psi} \mathbb{E}_{\tau \sim \Pi} \mathbb{E}_{(\mathcal{D}_\tau, \mathbf{x}^\star, y^\star)} \Big[-\log f_\psi(y^\star = k \mid \mathbf{x}^\star, \mathcal{D}_\tau \setminus \{(\mathbf{x}^\star, y^\star)\}) \Big].$$

Practice. Strong small-n performance, little tuning, fast inference; v2 strengthens priors, scaling, and calibration.(Hollmann et al., 2022, 2025)

TabPFN: handling distribution shift

Temporal and other shifts degrade IID assumptions. *Drift-Resilient TabPFN* encodes temporally evolving structural causal models in the prior and trains PFN to be robust to shifts, improving ID and OOD accuracy and calibration.(Helli et al., 2024)

- Prior Π becomes a stochastic process over parameters to simulate drift.
- Empirically outperforms XGBoost, CatBoost, and vanilla TabPFN under wild-time shifts.

TabICL: problem and idea

Challenge. Alternating full row/column attention becomes expensive when n is large. **Idea.** Pretrain on synthetic datasets with up to 60k samples and build *fixed-dimension row embeddings*, then run ICL over those embeddings for scalability.(Qu et al., 2025)

- Two-stage transformer: (1) column-then-row to embed rows, (2) ICL transformer over a compact set of row embeddings.
- Can handle up to 500k samples at inference on affordable hardware while maintaining ICL benefits.

TabICL: two-stage computation - 1

Let $\mathbf{x} \in \mathbb{R}^F$ be a row with mixed types. Stage 1 produces a fixed-dimensional row embedding

$$r(\mathbf{x}) \in \mathbb{R}^d, \qquad r(\mathbf{x}) = \underbrace{\text{RowTransformer}\big(\,\text{ColEmbed}(\mathbf{x})\,\big)}_{\text{inter-feature interactions within the row}},$$

where:

- **ColEmbed** is a *distribution-aware column-wise embedding*: for each feature j, a Set Transformer operates on that column's values across rows (training rows as K, V to avoid leakage) to produce a per-cell feature embedding for $\mathbf{x}_j$.
- **RowTransformer** is a transformer *across the F feature embeddings of the same row* (not across rows). It prepends a small number of learnable [CLS] tokens and uses RoPE to prevent representation collapse when feature distributions are similar. The concatenated [CLS] outputs form $r(\mathbf{x})$.

TabICL: two-stage computation - 2

Stage 2 (dataset-wise ICL). For a support set $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$ and a query $\mathbf{x}^\star$,

$$\mathcal{S} = \big[\, r(\mathbf{x}_1),\, e(y_1),\, \ldots,\, r(\mathbf{x}_m),\, e(y_m),\, r(\mathbf{x}^\star),\, e(\texttt{[MASK]}) \,\big],$$

and a causal-masked transformer outputs logits at `[MASK]`:

$$o = W_o\, h_{\texttt{[MASK]}} + b, \qquad p_\psi(y \mid \mathbf{x}^\star; S) = \mathrm{softmax}(o).$$

Training minimizes the cross-entropy at the `[MASK]` position over synthetic tasks. (Qu et al., 2025)

TabICL: Concrete Example - 1

Target x^*: Region=`New_Region`, Vehicle=`SUV`, DriverAge=`24`, BonusMalus=`1.1`

Retrieve $K=3$ **neighbors (by CLS embedding cosine)**

ID	Region	Vehicle	DriverAge	ClaimCount
R1	RegionA	SUV	25	1
R2	RegionB	SUV	23	0
R3	RegionA	Crossover	24	1

ICL prompt tokens (conceptual):

- $[(x^{\mathrm{R1}}, y^{\mathrm{R1}}{=}1), (x^{\mathrm{R2}}, y^{\mathrm{R2}}{=}0), (x^{\mathrm{R3}}, y^{\mathrm{R3}}{=}1),\ x^*]$
- Causal/self-attention with target masked; outcomes only for context rows

TabICL: Concrete Example - 2

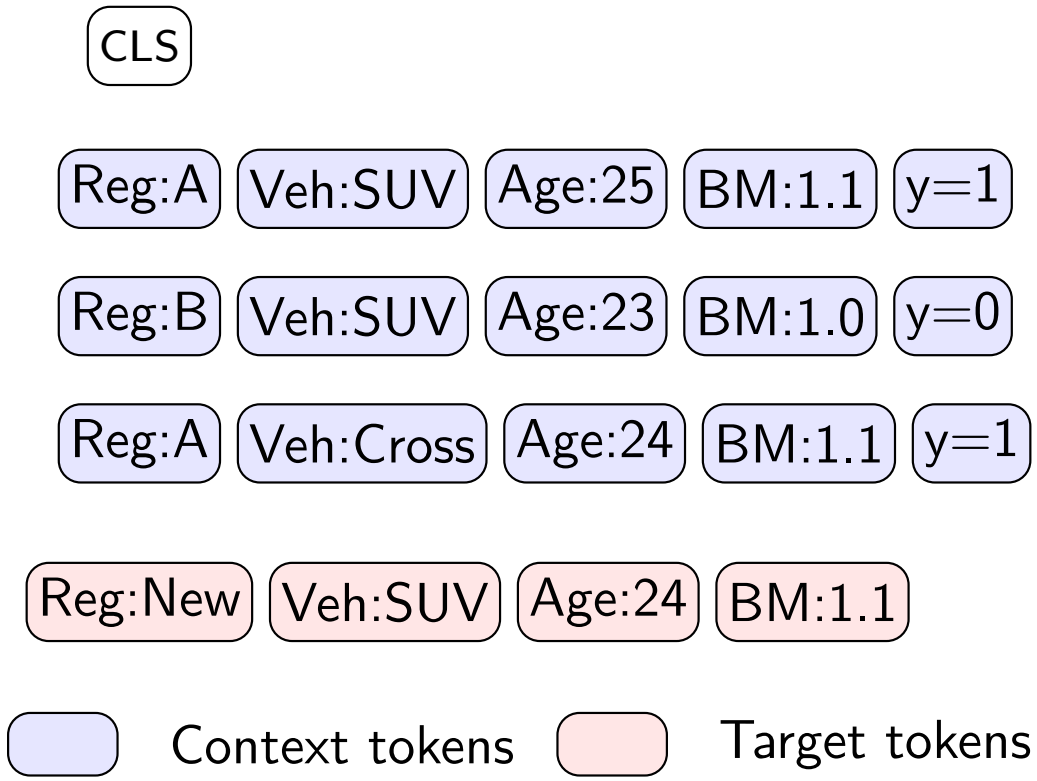
Prediction (frequency):

- Base CT (no context): $\hat{\mu}_{\text{base}}(x^*) = 0.072$
- TabICL (with context): $\hat{\mu}_{\text{ICL}}(x^* \mid \{\text{R1,R2,R3}\}) = 0.094$

Notes: Frozen decoder preserves calibration; ICL nudges representation toward similar risks; log retrieved IDs/similarities for audit.

TabICL: Tokenization & Decoration - 1

Sequence layout (conceptual):



TabPFN vs TabICL: when to use which

If $n \leq 10$k: TabPFN-v2 is a powerful default with excellent calibration and speed (Hollmann et al., 2025)

If n is large (tens of thousands to hundreds of thousands): TabICL often wins on accuracy and wall time (Qu et al., 2025)

If distribution shift is a concern: drift-aware TabPFN can be strong when the shift is encoded in the prior (Helli et al., 2024)

If latency and memory are tight: TuneTables yields small learned contexts for PFNs with strong results (Feuer et al., 2024)

The Challenge in Actuarial Modeling

- **Limited Data Problem**:
  - New products or regions
  - Rare events
  - New vehicle models
- **Traditional Solution**: Credibility Theory (Bühlmann, 1967)
  - Bühlmann framework
  - Linear combination of individual and collective experience
- **Modern Challenge**:
  - Complex non-linear patterns
  - High-dimensional feature spaces
  - Need for dynamic adaptation

Evolution: From Credibility to ICL

- **Classical Credibility (Bühlmann, 1967)**: Linear blend of individual and collective experience
- **Credibility Transformer (Richman, Scognamiglio, & Wüthrich, 2025)**:
  - Embeds credibility in attention mechanism
  - CLS token as learnable prior
- **ICL-Enhanced CT (This Work)**:
  - Dynamic context from similar instances
  - Zero-shot generalization capability
  - No retraining required for ICL!
  - Padayachy, Richman, Scognamiglio, and Wüthrich (2025)

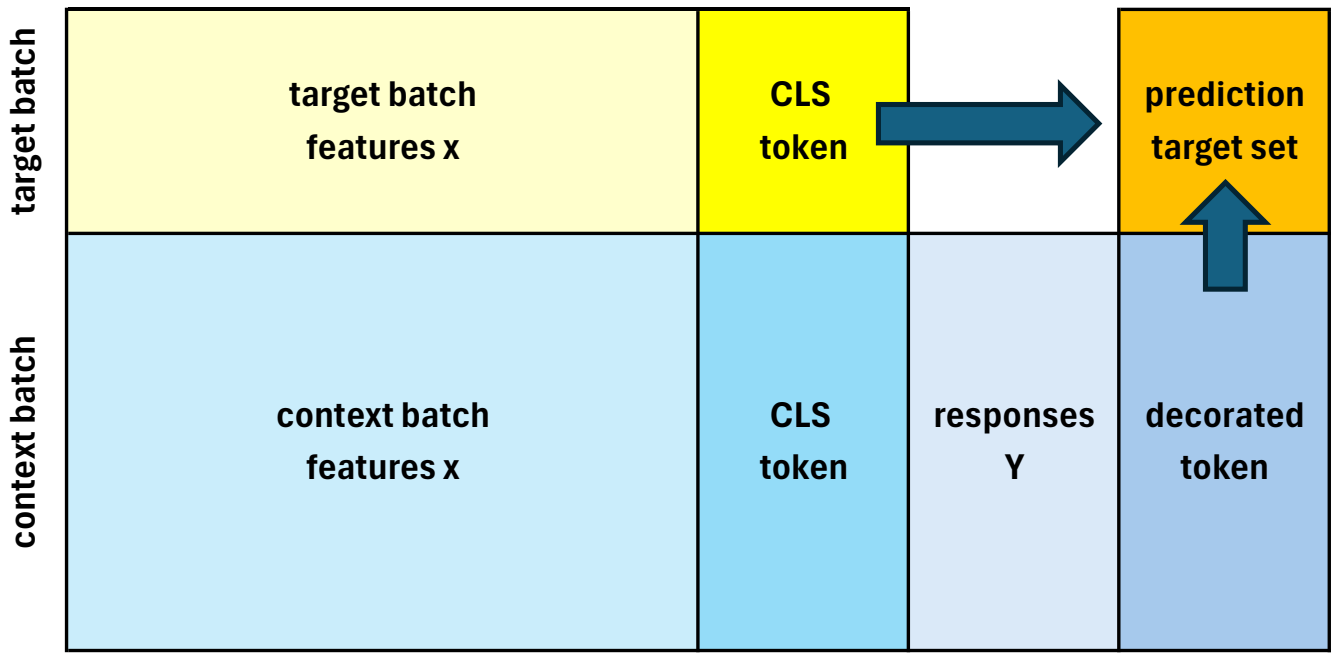
In-Context Learning: Key Innovation

- **ICL in Actuarial Context**:
  - Context = similar historical policies
  - Adaptation = adjusting predictions based on context
  - Zero-shot = handling new risk profiles

Key questions

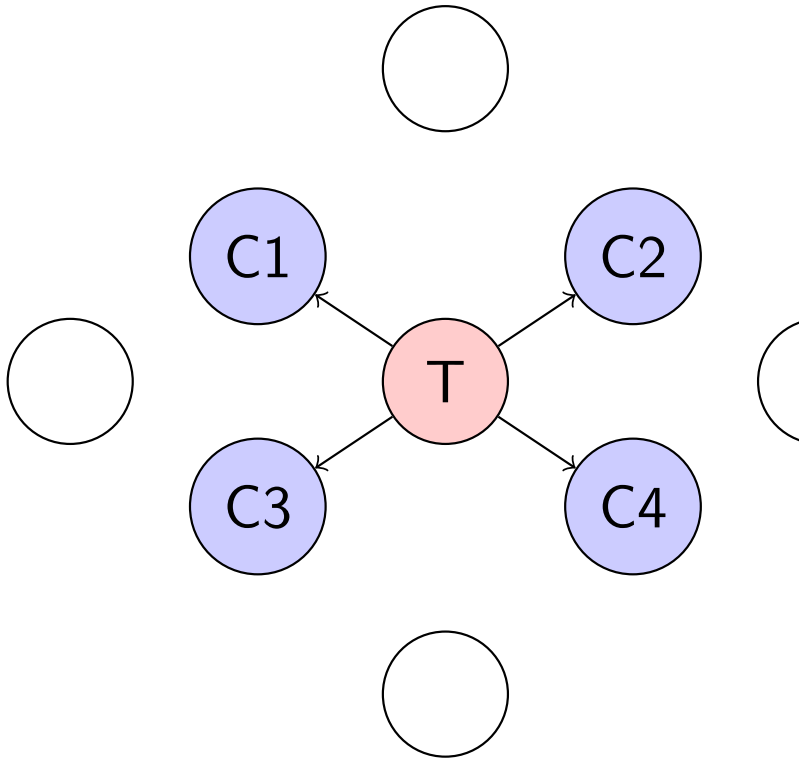
- Can ICL improve performance on a model that has been pre-trained using supervised learning?
- If this is the case, what is the explanation for this improvement?
- Can ICL for tabular data be used to improve the performance of a much smaller pre-trained model focussing only on a single dataset?

ICL-CT: Architecture Overview



Component 1: Context Retrieval

- **Purpose**: Retrieve similar risks for context
- **Space**: Base CT CLS embeddings (ℓ_2-normalized)
- **Metric**: Cosine similarity (inner product)
- **Retrieval**: $K = 64$ neighbors per target; union across chunk
- **Batching**: Keep top $c = 1000$ context; target chunk size $m = 200$



Context retrieval in embedding space

Component 2: Outcome Token Decorator

- **Purpose**: Inject observed outcomes from the context into their CLS tokens in a credibility-weighted way.
- **Definition** (context j):

$$\boldsymbol{c}^{\mathrm{decor}}(\boldsymbol{x}_j) = \widehat{\boldsymbol{c}}^{\mathrm{cred}}(\boldsymbol{x}_j) + \frac{v_j}{v_j + \kappa}\, \boldsymbol{z}^{\mathrm{FNN1}}(Y_j).$$

- **Notes**:
  - Applied to context only; targets keep $\widehat{\boldsymbol{c}}^{\mathrm{cred}}(\boldsymbol{x}_i)$ (no outcomes).
  - $\boldsymbol{z}^{\mathrm{FNN1}}(\cdot)$ is a learned embedding of the response; exposures v enter only via $\frac{v}{v+\kappa}$ to avoid leakage.

Component 3: Causal Self-Attention

- **Setup**: Concatenate [context | target] and apply causal mask $\boldsymbol{M}^{\infty}$ to block target–target links.
- **Q/K/V**: Time-distributed FNNs on tokens:
  - Context: from $\boldsymbol{c}^{\mathrm{decor}}$ (depends on Y).
  - Target: from $\widehat{\boldsymbol{c}}^{\mathrm{cred}}$ (feature-only).
- **Causal attention**:

$$\boldsymbol{A} = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\top}}{\sqrt{2b}} + \boldsymbol{M}\right), \quad \boldsymbol{H} = \boldsymbol{A}\,\boldsymbol{V} \tag{1}$$

- **Effect**: Propagates outcome-enriched context information to target CLS tokens via attention weights $a_{i,j}$.

Component 4: Frozen Decoder and Output

- **Decoder**: Use frozen decoder from base CT
- **Prediction on targets**:

$$\widehat{\mu}^{\text{ICL-CT}}(\boldsymbol{x}_i; \mathcal{B}_{\text{context}}) \;=\; \widehat{\boldsymbol{z}}^{\text{decod}}\Big(\boldsymbol{c}_i^{\text{ICL-trans}}\Big), \quad i \in \mathcal{I}_{\text{target}} \tag{2}$$

- **Benefits**: Preserves calibration; regularizes ICL adjustments

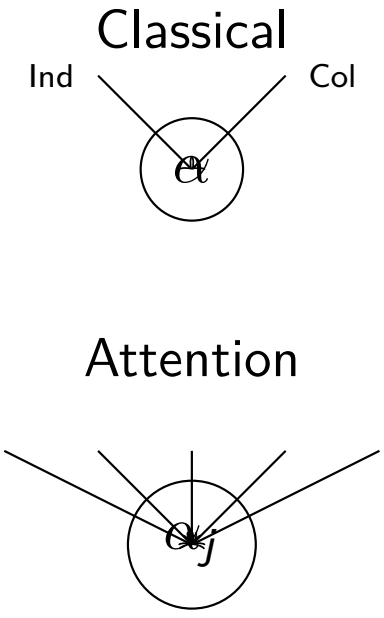
Attention as Generalized Credibility

- **Attention-based Credibility**:

$$\hat{\mu} = \sum_j \alpha_j(\boldsymbol{x}) \cdot \boldsymbol{v}_j \qquad (3)$$

where $\alpha_j(\boldsymbol{x})$ are attention weights

- **Advantages**:
  - Feature-dependent weights
  - Multiple information sources
  - Non-linear combinations



Proposition: Credibility via Attention

Statement (paper Prop. 1): For target instance i, the causal attention head produces

$$\boldsymbol{h}_i \;=\; a_{i,i}\, \boldsymbol{z}_V^{\mathrm{FNN}}\Big(\widehat{\boldsymbol{c}}^{\mathrm{cred}}(\boldsymbol{x}_i)\Big) \;+\; \sum_{j\in\mathcal{I}_{\mathrm{context}}} a_{i,j}\, \boldsymbol{z}_V^{\mathrm{FNN}}\Big(\widehat{\boldsymbol{c}}^{\mathrm{cred}}(\boldsymbol{x}_j) + \tfrac{v_j}{v_j+\kappa}\, \boldsymbol{z}^{\mathrm{FNN1}}(Y_j)\Big)\,,$$

with $a_{i,j} \geq 0$ and $a_{i,i} + \sum_{j\in\mathcal{I}_{\mathrm{context}}} a_{i,j} = 1$, and $a_{i,j}{=}0$ for j in other targets (by masking).

Interpretation: A credibility blend between the target's own signal and context signals enriched by outcomes with weight $\frac{v}{v+\kappa}$.

Proof Sketch

- Causal mask $\boldsymbol{M}^{\infty}$ zeros target–target interactions, leaving self and context terms only.
- Softmax over $\boldsymbol{Q}\boldsymbol{K}^{\top}/\sqrt{2b} + \boldsymbol{M}$ yields normalized nonnegative weights $a_{i,j}$ on $\{i\} \cup \mathcal{I}_{\text{context}}$.
- Attention head computes $\boldsymbol{h}_i = \sum_j a_{i,j}\, \boldsymbol{v}_j$ with values built from decorated tokens for context and plain cred CLS for the target, giving the stated credibility structure.

Linearized ICL Variant

Idea: Make the attention weights independent of outcomes by using feature-only queries/keys.

$$\widetilde{\boldsymbol{Q}} = \boldsymbol{z}_Q^{\mathrm{FNN}}\big(\boldsymbol{c}^{\mathrm{cred}}\big), \quad \widetilde{\boldsymbol{K}} = \boldsymbol{z}_K^{\mathrm{FNN}}\big(\boldsymbol{c}^{\mathrm{cred}}\big), \quad \boldsymbol{V} = \boldsymbol{z}_V^{\mathrm{FNN}}\big(\boldsymbol{c}^{\mathrm{decor}}\big).$$

- **Effect**: Predictions become linear in Y through $\boldsymbol{V}$, while $\widetilde{\boldsymbol{Q}}, \widetilde{\boldsymbol{K}}$ depend only on features.
- **Caveat**: Guarantees hold cleanly for a single ICL layer; deeper stacks may reintroduce non-linearities via intermediate transformations.
- **Empirics**: Linearized model slightly underperforms the 2-layer non-linear ICL prior to joint fine-tuning but closes the gap after.

Three-Phase Training

1. **Phase 1: Base CT pretraining**
   - AdamW (LR 10^{-3}, WD 10^{-2}, β_2=0.95), batch 1024
   - Poisson deviance; early stopping (patience 20)
2. **Phase 2: ICL fine-tuning**
   - Insert decorator + 2 ICL layers; freeze decoder
   - AdamW (LR $3 \cdot 10^{-4}$, WD 10^{-2}, β_2=0.95)
   - Causal mask; loss on target rows only
3. **Phase 3: Joint fine-tuning**
   - Unfreeze all; AdamW (LR $3 \cdot 10^{-5}$)
   - Early stopping (patience 10)

Training Procedure

ICL-CT training

- Form batches as $[\mathcal{B}_{\text{context}} \,\|\, \mathcal{B}_{\text{target}}]$; causal mask prevents target–target interactions
- Provide outcomes only for context; decorate tokens; apply ICL layers
- Loss applied to target rows (Poisson deviance)
- Inference uses retrieval procedure from Context Retrieval

Main Results (Conventional Split)

- **Base CT (single run)**: OOS Poisson deviance 23.743; original CT benchmark 23.788 ± 0.040.
- **ICL-CT (2 layers, decoder frozen)**: OOS 23.725.
- **ICL-CT (2 layers, fine-tuned)**: OOS 23.710 (best single-run).
- **Ensembled (5 runs)**: 2-layer OOS 23.679 (pre-FT), 23.676 (post-FT).

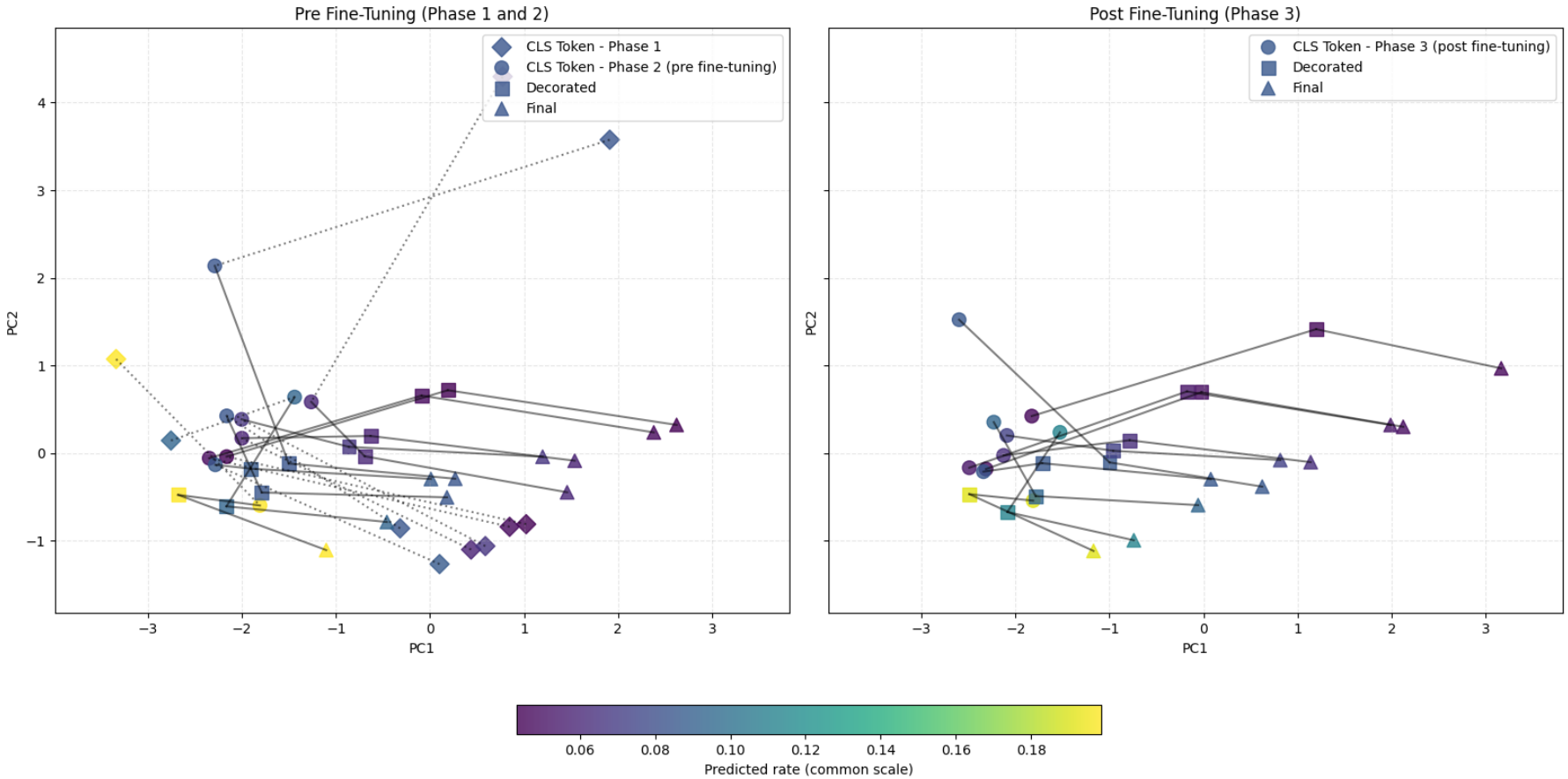
Units: 10^{-2} Poisson deviance.

Neighborhood Dynamics

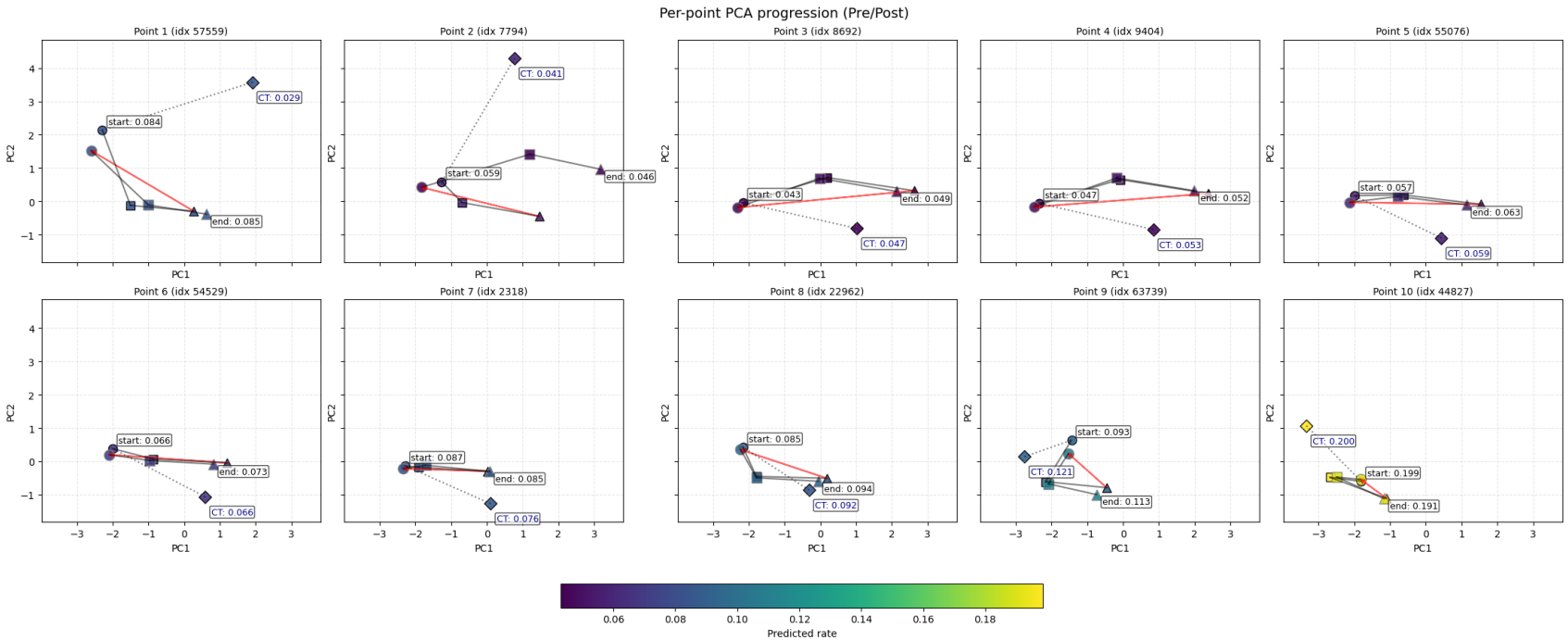
Distance metric: Cosine similarity on ℓ_2-normalized CLS embeddings; rank by best match per candidate.

- Tightening: outcome decoration tightens neighborhoods (closest distances drop by 10–40%).
- Selective broadening: final ICL admits near-but-diverse neighbors while preserving key covariates.
- Cohesion: decoration amplifies coherence by fuel/region; pulls exact brand matches into top sets.
- Sparse slices: largest gains where combinations are rare.

PCA Analysis of CLS Tokens



PCA Progression by Points



Zero-Shot Setup

- **Goal**: Evaluate generalization to unseen region categories
- **Test set**: Regions totaling 10% exposure remapped to *unseen*
- **Training**: Additional small-exposure regions remapped to *unseen*
- **Mechanism**: Context retrieved from training distribution only

Zero-Shot Data Split

Characteristic	**Training set**	**Test set**
Number of policies	601,781	76,226
Number set to unseen	165,200	76,226
Total exposure (years)	323,458	34,900
Number of claims	24,006	2,377
Average frequency	7.42%	6.81%

Zero-Shot Results (Unseen Regions)

- **Null model**: OOS 21.091 (baseline).
- **Base CT (phase 1)**: OOS 20.282.
- **ICL-CT (2 layers, phase 2)**: OOS 20.264.
- **ICL-CT (2 layers, phase 3)**: OOS 20.259 (best).

Units: 10^{-2} Poisson deviance. Results per Table in paper's zero-shot section.

Key Takeaways

- FMs provide scalable priors; adapt with prompting, PEFT, retrieval
- GPT series unlocked few-shot and zero-shot CoT; reasoning improves with scale and cues
- Tabular FMs: TabTransformer/FT-Transformer/TransTab; TabPFN for small-N
- TabICL: fast Bayesian-flavored adaptation with context
- ICL-CT: integrates credibility with ICL, improves robustness and calibration

References I

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... others (2020). Language models are few-shot learners. In *Advances in neural information processing systems.*

Bühlmann, H. (1967). Experience rating and credibility. *ASTIN Bulletin*, *4*(3), 199–207.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacl-hlt.* Retrieved from https://arxiv.org/abs/1810.04805

Feuer, B., Schirrmeister, R. T., Cherepanova, V., Hegde, C., Hutter, F., Goldblum, M., ... White, C. (2024). *Tunetables: Context optimization for scalable prior-data fitted networks.* Retrieved from https://arxiv.org/abs/2402.11137 (NeurIPS 2024 Poster)

References II

Garg, S., Tsipras, D., Roelofs, R., Hazan, E., et al. (2022). What can transformers learn in-context? a case study of simple function classes. *arXiv preprint arXiv:2208.01066*. Retrieved from https://arxiv.org/abs/2208.01066

Geva, M., et al. (2021). Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2110.02834*. Retrieved from https://arxiv.org/abs/2110.02834

Gorishniy, Y., Rubachev, I., Khrulkov, V., & Babenko, A. (2021). Revisiting deep learning models for tabular data. *arXiv preprint arXiv:2106.11959*. Retrieved from https://arxiv.org/abs/2106.11959

Helli, K., Schnurr, D., Hollmann, N., Müller, S., & Hutter, F. (2024). *Drift-resilient tabpfn: In-context learning temporal distribution shifts on tabular data.* Retrieved from https://arxiv.org/abs/2411.10634 (NeurIPS 2024)

References III

Hoffmann, J., Borgeaud, S., Mensch, A., et al. (2022). Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*. Retrieved from https://arxiv.org/abs/2203.15556

Hollmann, N., Müller, S., Eggensperger, K., & Hutter, F. (2022). *Tabpfn: A transformer that solves small tabular classification problems in a second.* Retrieved from https://arxiv.org/abs/2207.01848

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., ... Hutter, F. (2025). Accurate predictions on small data with a tabular foundation model. *Nature*, *637*(8045), 319–326. doi: 10.1038/s41586-024-08328-6

Huang, X., Khetan, A., Cvitkovic, M., & Karnin, Z. (2020). Tabtransformer: Tabular data modeling using contextual embeddings. *arXiv preprint arXiv:2012.06678*. Retrieved from https://arxiv.org/abs/2012.06678

References IV

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., et al. (2020). Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*. Retrieved from https://arxiv.org/abs/2001.08361

Kojima, T., Sagawa, S., Lu, M. D., et al. (2022). Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Min, S., Lewis, M., Zettlemoyer, L., & Hajishirzi, H. (2022). Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*. Retrieved from https://arxiv.org/abs/2202.12837

Müller, S., Hollmann, N., Pineda Arango, S., Grabocka, J., & Hutter, F. (2021). *Transformers can do bayesian inference.* Retrieved from https://arxiv.org/abs/2112.10510

References V

Olsson, C., Elhage, N., Nanda, N., et al. (2022). *In-context learning and induction heads.* Technical report. (Available at https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html)

OpenAI. (2023). GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., et al. (2022). Training language models to follow instructions with human feedback. In *Advances in neural information processing systems.*

Padayachy, K., Richman, R., Scognamiglio, S., & Wüthrich, M. V. (2025). *In-context learning enhanced credibility transformer.* Retrieved from https://arxiv.org/abs/2509.08122

Qu, J., Holzmüller, D., Varoquaux, G., & Le Morvan, M. (2025). *Tabicl: A tabular foundation model for in-context learning on large data.* Retrieved from https://arxiv.org/abs/2502.05564 (ICML 2025)

References VI

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving language understanding by generative pre-training.* OpenAI Technical Report.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language models are unsupervised multitask learners.* OpenAI Technical Report.

Richman, R., Scognamiglio, S., & Wüthrich, M. V. (2025). The credibility transformer. *European Actuarial Journal*. (Forthcoming)

Su, J., Lu, Y., Pan, S., Wen, B., & Liu, Y. (2021). Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*. Retrieved from `https://arxiv.org/abs/2104.09864`

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems.* Retrieved from `https://arxiv.org/abs/1706.03762`

References VII

Wang, X., Wei, J., Schuurmans, D., et al. (2022). Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wang, Z., & Sun, J. (2022). Transtab: Learning transferable tabular transformers across tables. *arXiv preprint arXiv:2205.09328*. Retrieved from `https://arxiv.org/abs/2205.09328`