Predicting Training Re-evaluation Curves Enables Effective Data Curriculums for LLMs

(Training data retention, as a function of training step, is predictable)

Shane Bergsma, **Nolan Dey, Joel Hestness**

Cerebras Systems

April 16, 2026

When Should We Train on High-Quality Data?

Mid-training/annealing phases now common in LLM pre-training.

- But what *principles* govern optimal data placement?

We introduce the **training re-evaluation curve** (**TREC**):

- TREC measures how well each training batch is remembered by the final model.

Key Finding: Place high-quality training data at *low points on TREC* → get *better performance* on held-out high-quality data (after retraining)

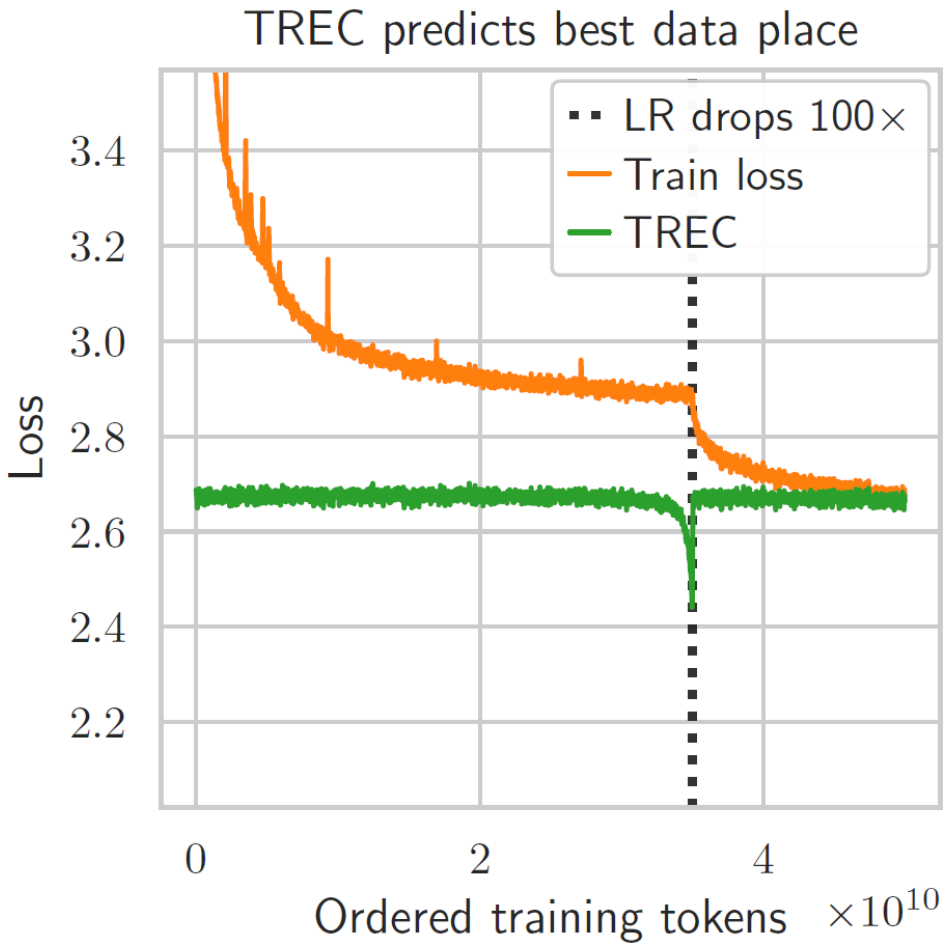
Crucially: The TREC is *predictable in advance* from optimizer settings.

🎯 Use it to *design effective data curriculums,* before training.

Training Data Is Not Retained Equally

TREC: evaluates training batches (in order) using final (frozen) model weights

- **Train loss**: steadily falls on (new) batches (**orange curve**), especially after learning rate (LR) drops 100x (dashed line ⋯⋯)
- **TRECs**: data is retained best right *before* the LR drop (**green curve**)

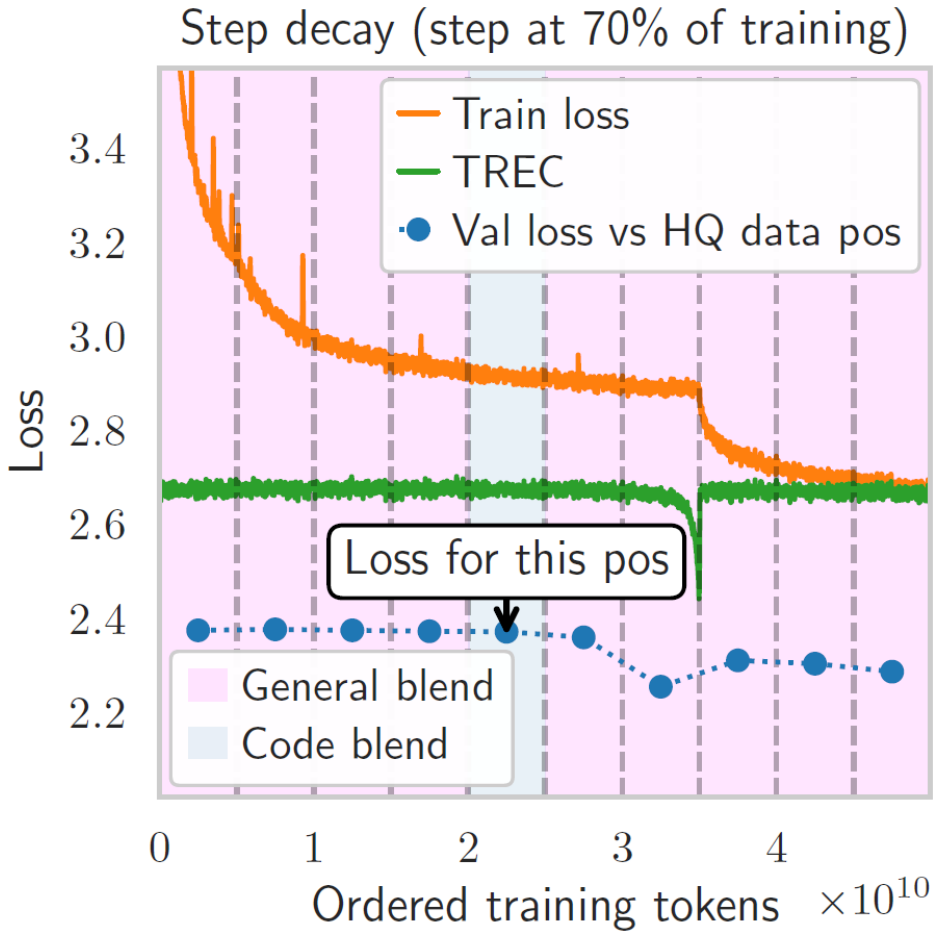


TREC Predicts Optimal Data Placement

Placement tests: Ten experiments, running end-to-end training but with high-quality "Code blend" used in a different 10% segment.

Finding: Lowest validation loss (on code data) corresponds to segment with lowest TREC loss: low TREC == better placement (**blue dots**)

- Finding recurs with 10% and decay-to-zero learning rate schedules, at 610M and 3.9B scales, and aligns with results in prior work.



Two Ratios + Learning Rate Schedule Govern TREC Shape

Tokens-per-parameter

$$\mathrm{TPP} = \frac{D}{N}$$

D: Number of training tokens

N: Number of model parameters

AdamW Timescale

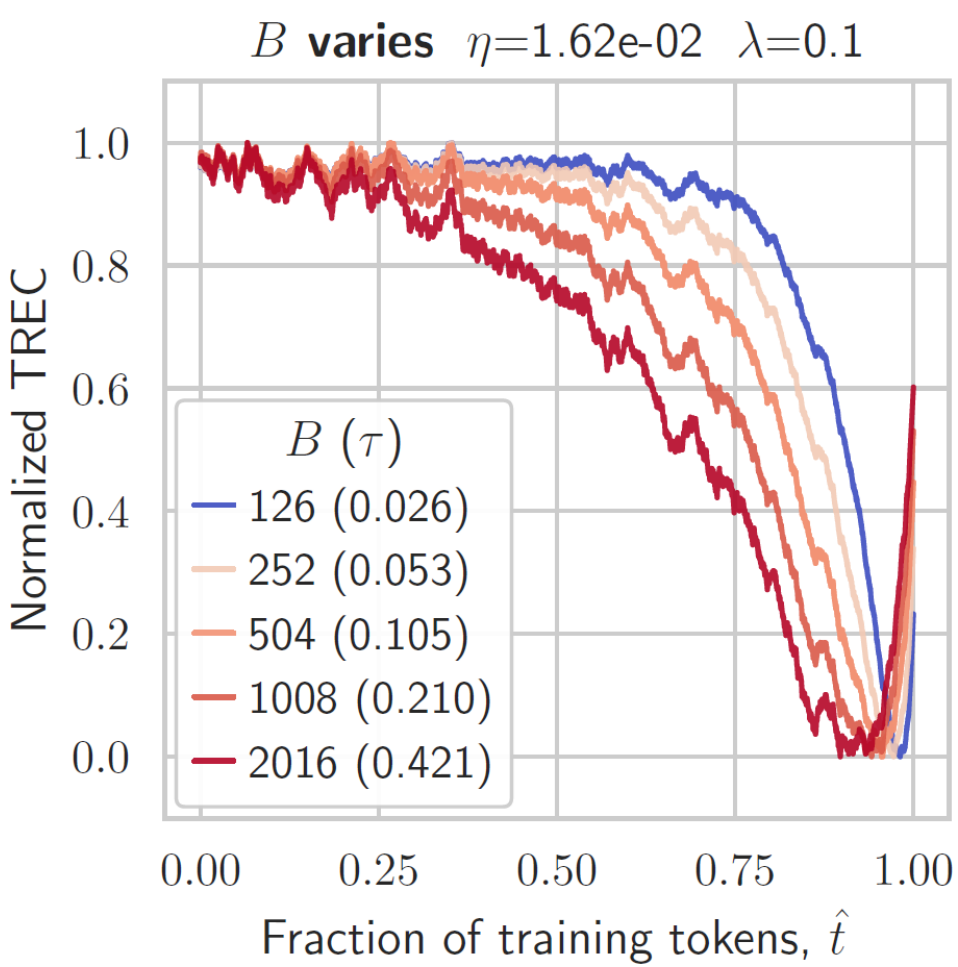
$$\tau = \frac{B}{\eta \lambda D}$$

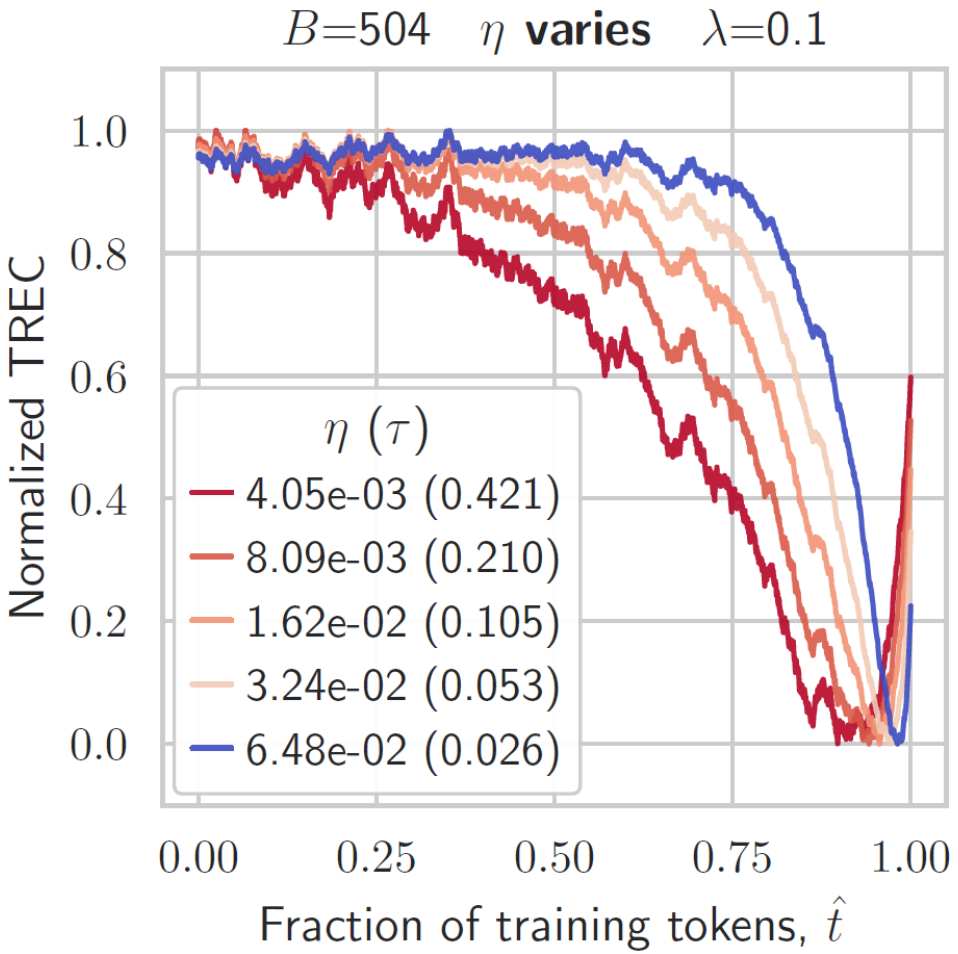
B: Batch size in tokens

η: Learning rate λ: Weight decay

Two Ratios Govern TREC Shape

$$\tau = \frac{B}{\eta \lambda D}$$

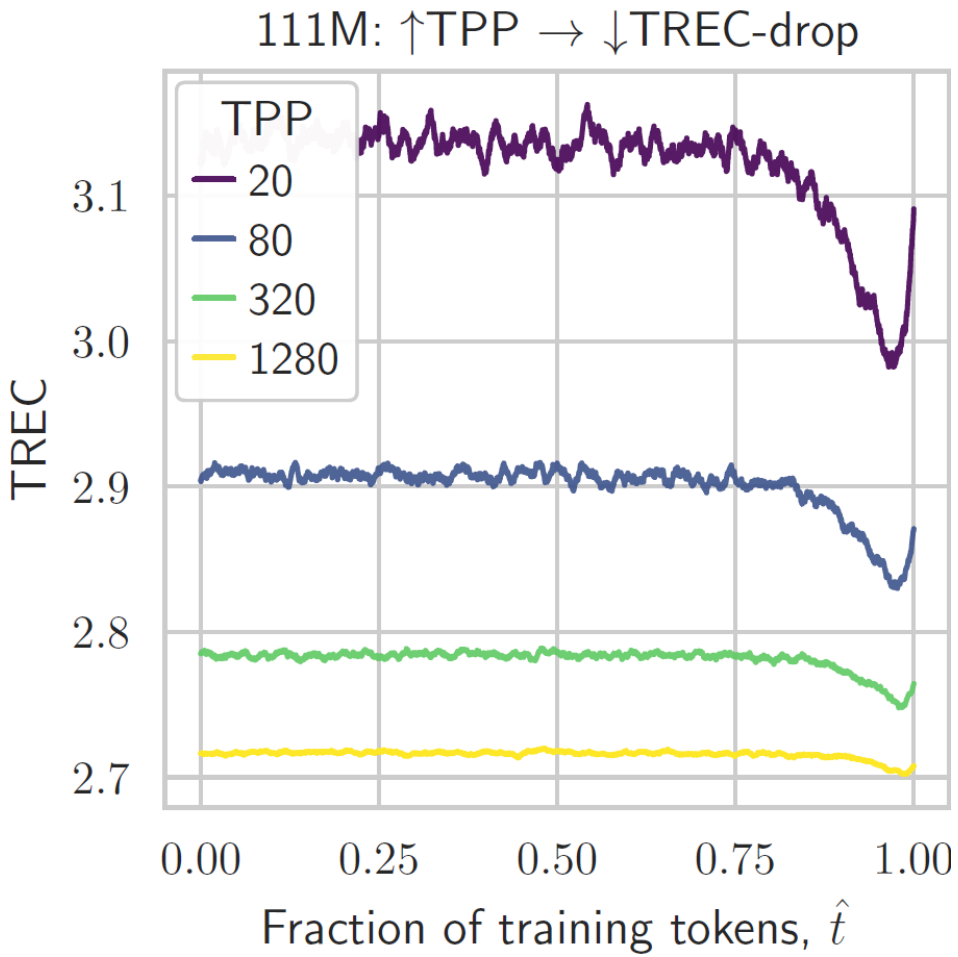


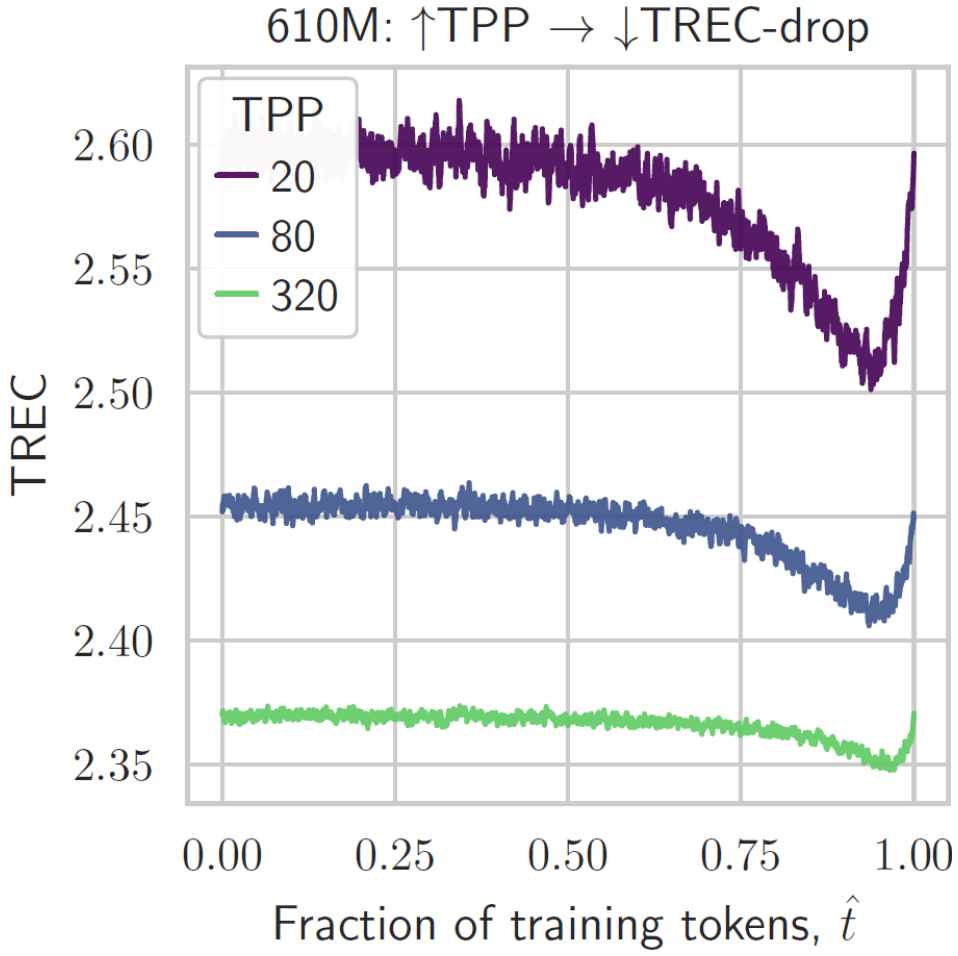


Key finding: TRECs align when τ matches

Two Ratios Govern TREC Shape

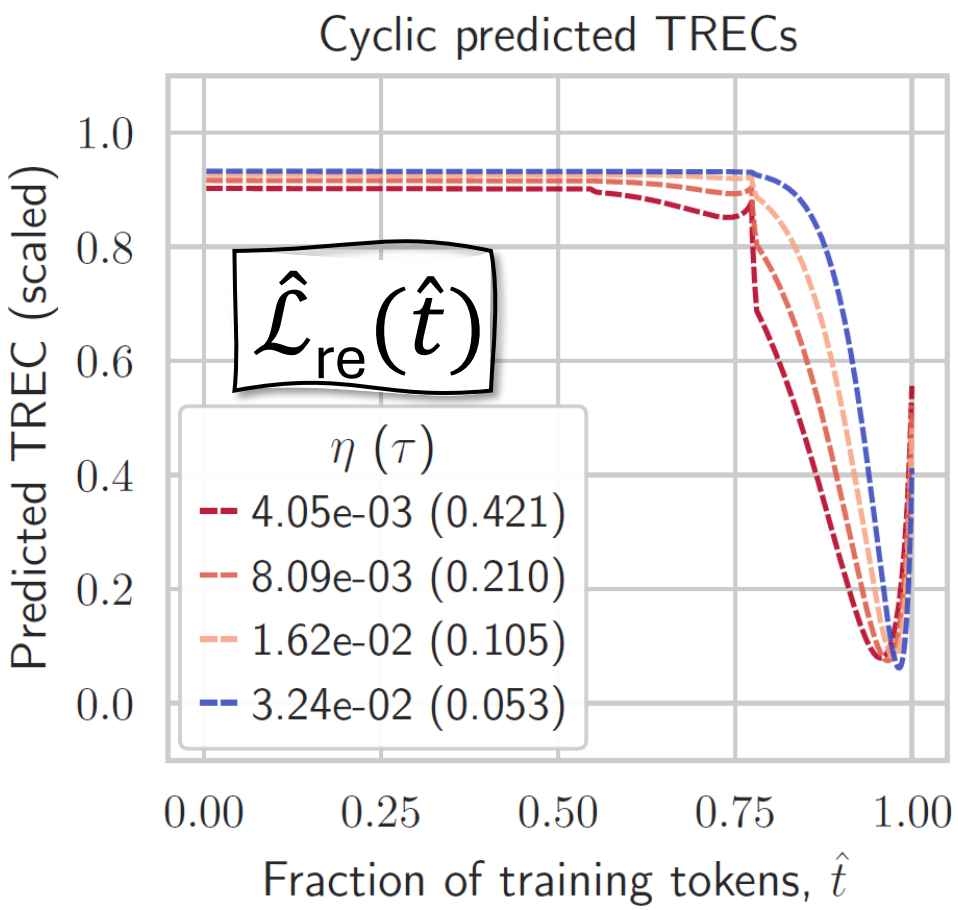
$$\text{TPP} = \frac{D}{N}$$

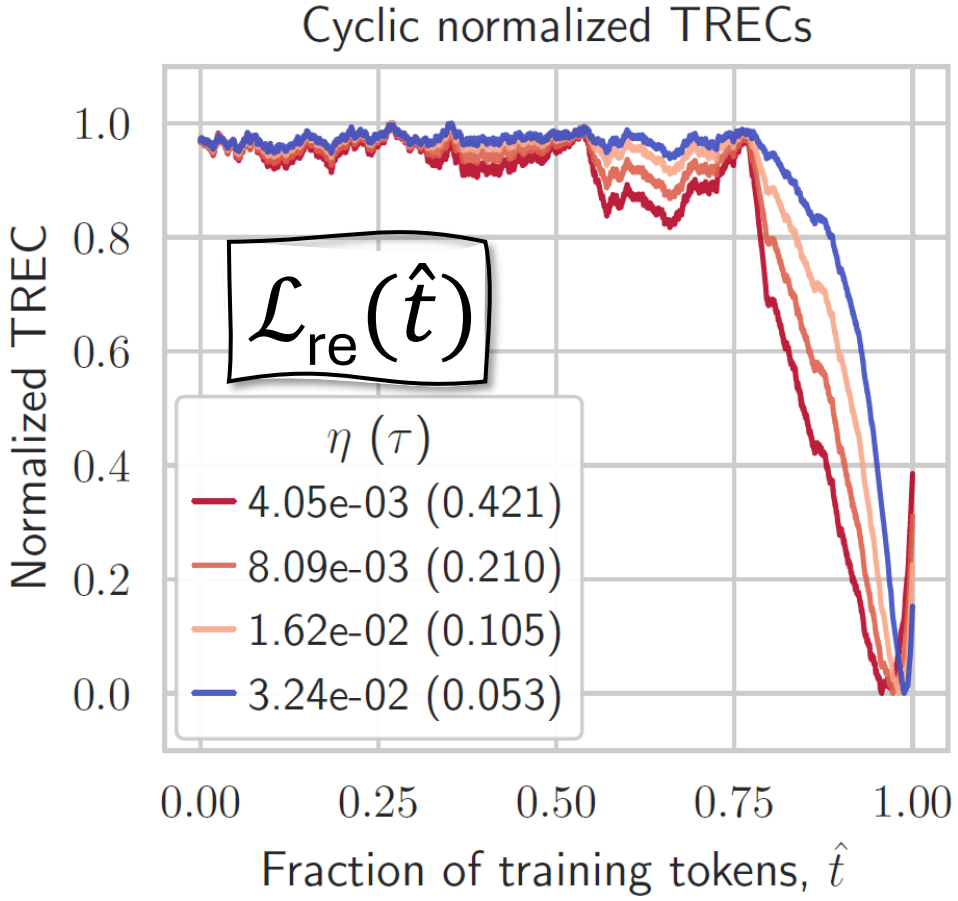




Key findings: TRECs align when TPP matches; bigger TPP → smaller TREC drop

Use TREC to Design Data Curricula





Key finding: TRECs are predictable