

TECHNICAL TALK

Weak-to-Strong Generalization for Better Retrieval and Reranking

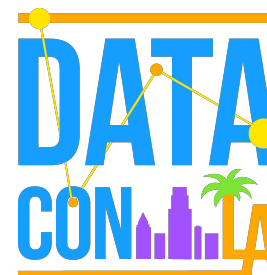
By Devyn Miller





What This Talk Covers

- Introduction/Primer
- Why cross-entropy against weak labels produces a ceiling
- Weak-to-strong generalization and auxiliary confidence loss
- Pipeline: BM25 + MiniLM → large cross-encoder
- Failures and fixes
- Limitations



PRIMER

Before We Dive In...

Definitions and general concepts



What Is Retrieval?

- 1 Given a query, find the most relevant documents
- 2 Out of possibly millions of candidates
- 3 Ranked from best match to worst match
- 4 Powers search engines and RAG chatbots alike

RAG = “Retrieval-Augmented Generation”; an AI assistant looking something up before it answers.





Two Stages: Retrieval, Then Reranking

STAGE 1 · RETRIEVAL

Fast & coarse

Millions of docs → shortlist of ~100–1,000



STAGE 2 · RERANKING

Slow & precise

Shortlist → a carefully ordered final list

Why split it in two? Stage 1 has to scan everything, so it must be cheap. Stage 2 only looks at the shortlist, so it can afford to be smart. This talk is about training Stage 2 well.



Lexical Matching vs. Semantic Understanding

BM25

"Best Matching" counts shared words

- Weights rare words more than common ones
- Fast enough to scan millions of docs
- "Car" and "automobile" look unrelated to it
- Zero understanding of meaning

Cross-Encoder

Reads query + doc together, attends across both

- Learned from massive pretraining
- Understands synonyms, paraphrase, context
- "Car" and "automobile" look related to it
- Too slow to run over millions of docs which is why it only sees the shortlist



How Models Learn: Labels + Loss



The catch: the loss function trusts the label completely, right or wrong, which is the root of the problem this talk solves.



Reading the Scoreboard

Metric	What it tells you
NDCG@10	Rewards getting the right documents near the top of the top 10
MRR@10	How fast you hit the first correct answer
Recall@k	Of all the relevant docs out there, how many did you surface at all
PGR	How much of the weak-to-strong gap you actually closed



Important Definitions

Weak signal

A cheap, imperfect source of labels, such as BM25, clicks, or a small model

Strong model

The bigger, more capable model we're training

Ceiling

The best possible performance; i.e., what you'd get with perfect, gold labels

Confidence

How sure the model states it is about its own guess

**Now... let's build a reranker that's
smarter than its own supervisor!**



01

PART 01

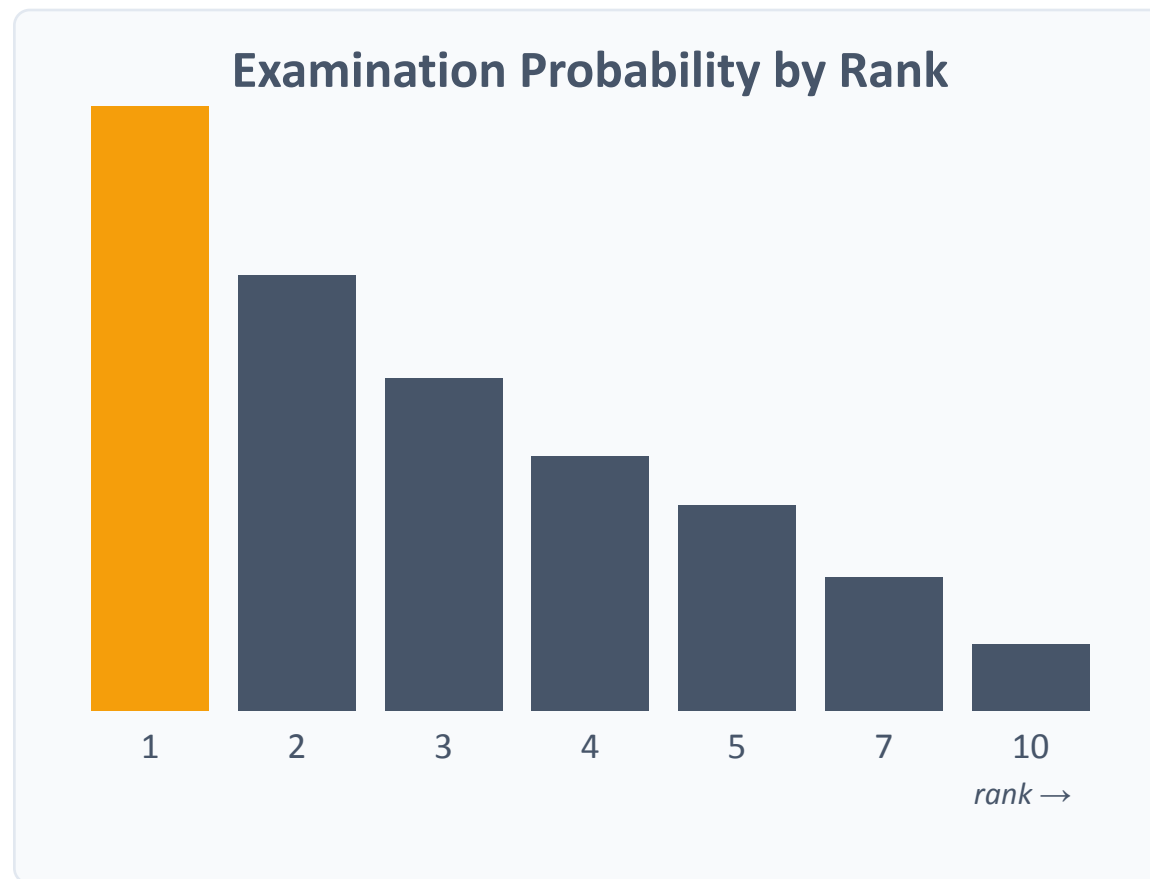
Why Weak Labels Cap Your Model



Click Logs Encode Examination



- **P(click) factors into two independent terms**
- $P(\text{examine} \mid \text{rank}) \cdot P(\text{relevant} \mid q, d)$
- Examination probability drops roughly 10x from rank 1 to rank 10
- Training on raw CTR fits the easier factor: position
- Offline nDCG improves against a biased evaluation set built from the same logs



THE PROBLEM

Every Cheap Label Has a Correlated Error Term

Signal	Systematic error
BM25	No synonymy, no paraphrase
Clicks	Position, presentation, and selection bias
Bi-encoder	Rare entities, exact identifiers
Small reranker (MiniLM)	Truncation at 256 tokens
Heuristics	Drift when content changes



CE Against a Weak Label Has a Known Minimizer

Loss has exactly one goal: make the student match the weak label... everywhere

A correct label

The labeler's mistake

Identical target to the loss function

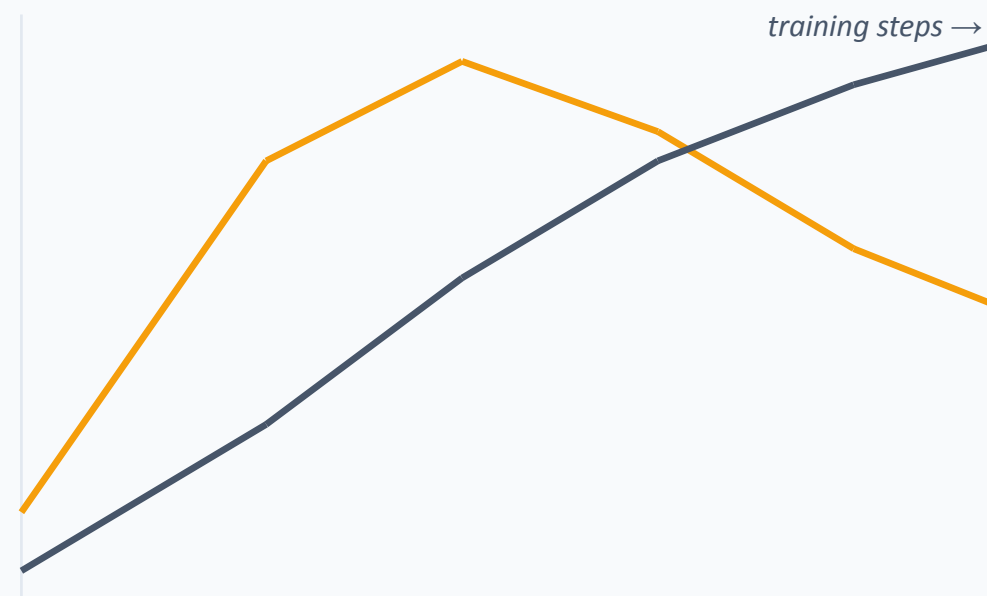
More capacity → closer to that match, not further from it.



Naive Fine-Tuning Still Exceeds the Labeler

- Weak errors inconsistent across inputs
- Models learn shared signals before noise
- Noise memorization: late in training
- Early stopping captures performance before memorization
- **Burns et al.: ~50% of gap recovered naively**

Training Dynamics



- gold nDCG (peaks, then falls)
- agreement with labeler (rises)

The Auxiliary Confidence Loss

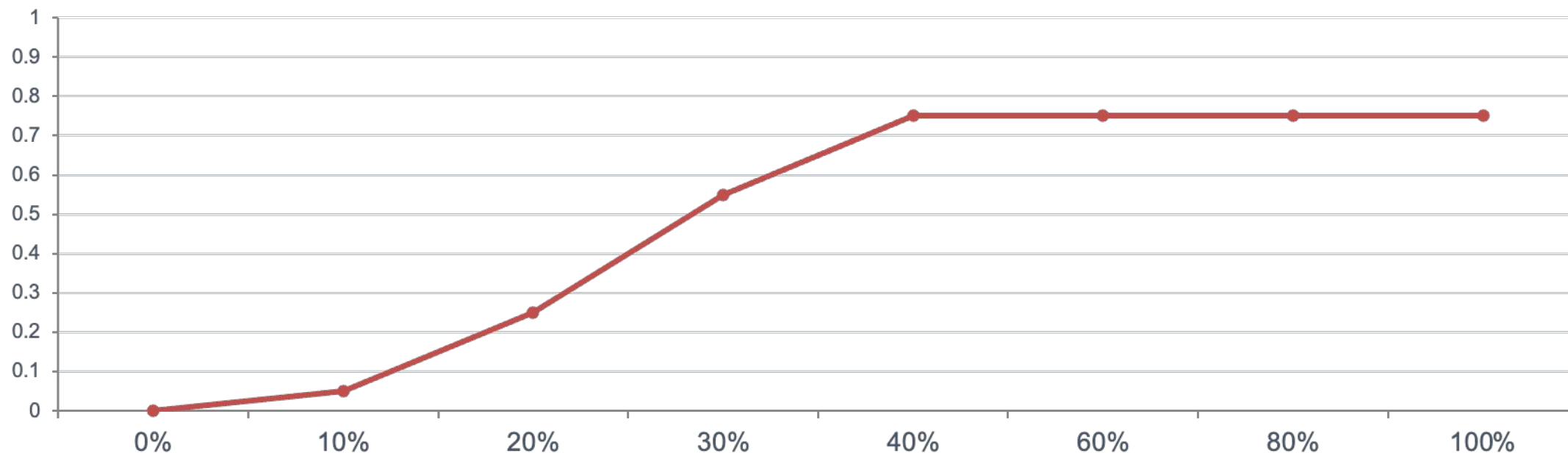


```
loss = (1 -  $\alpha$ ) * CE(weak_target, student_pred) +  $\alpha$  *  
CE(sharpen(student_pred).detach(), student_pred)  
  
# term 1: standard fine-tuning, weighted by reliability  
# term 2: cross-entropy to the model's own hardened output  
# detach() establishes target behavior
```

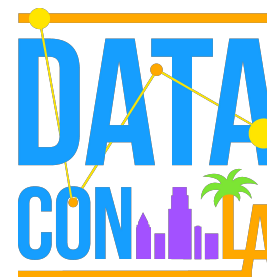
- $\alpha = 0$ reduces exactly to standard fine-tuning (generalization)



Alpha Controls How Much the Model Can Override

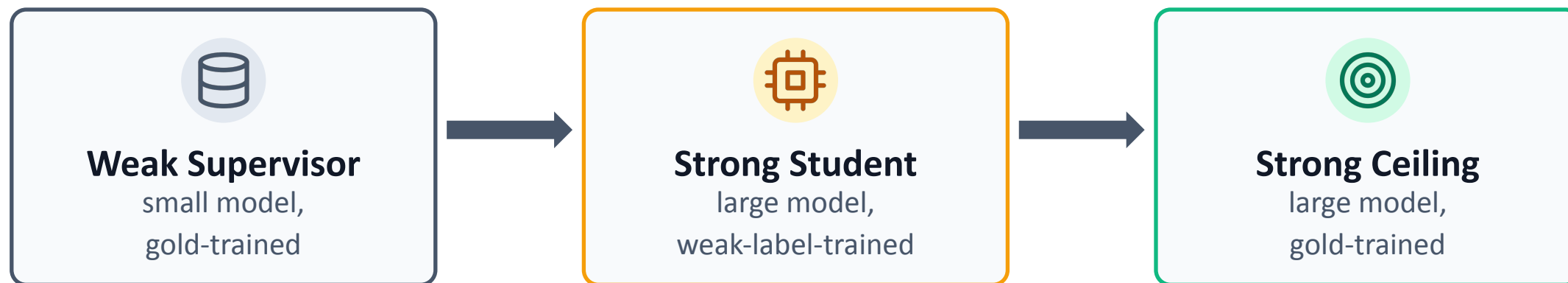


- % of training steps (x-axis) vs. α value (y-axis)
- Ramp must trail the task-learning phase



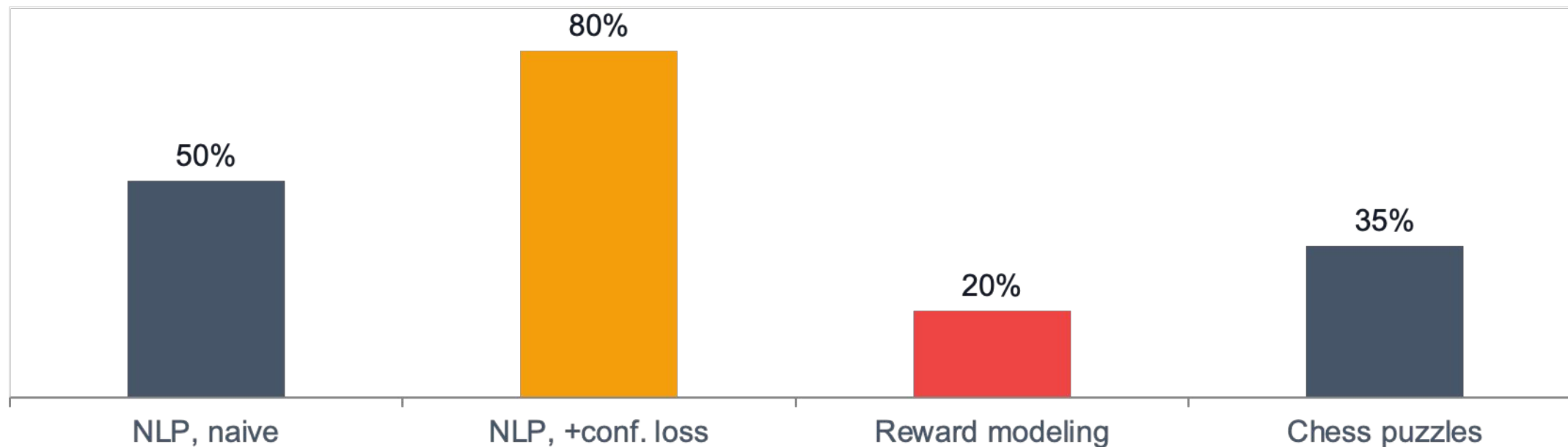


Burns et al., 2023 (arXiv:2312.09390)



- Motivation: humans eventually supervising models beyond human capability
- Structure maps onto BM25 → fine-tuned cross-encoder → gold-trained cross-encoder
- The alignment framing doesn't matter for using the mechanism

What the Paper Actually Found



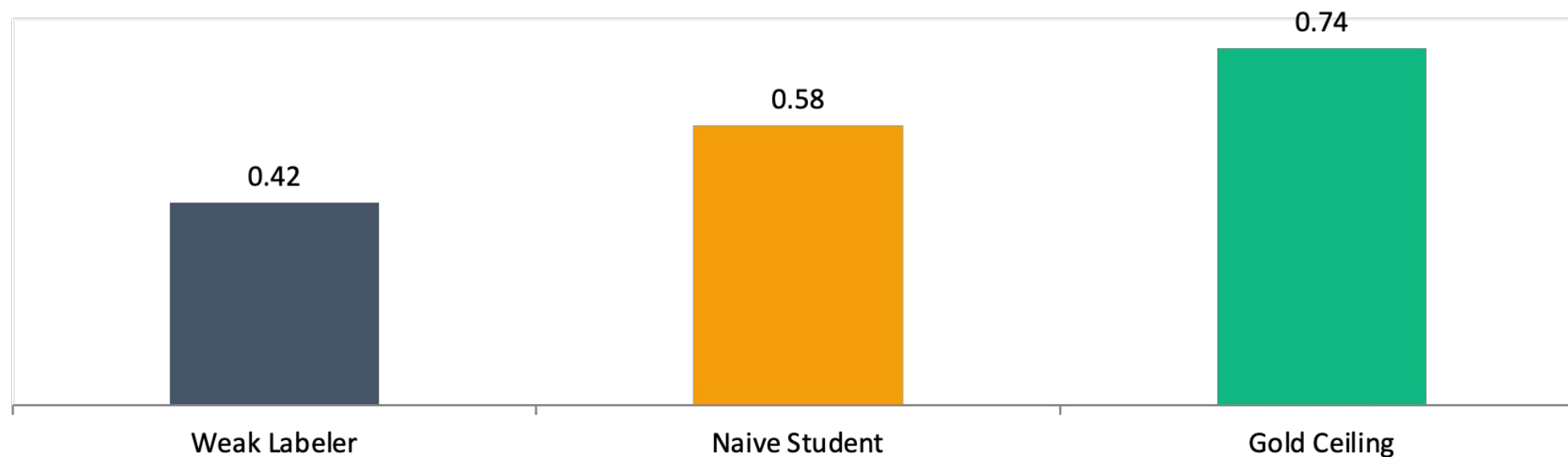
- Chess: bootstrapping through intermediate models helped; the confidence loss did not.



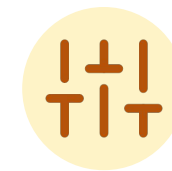
The gap from naive student to gold ceiling is unrecovered capability the model already has from pretraining.

THE MECHANISM

Three Reference Points Define the Problem



- The gap from naive student to gold ceiling is unrecovered capability the model already has from pretraining.



Add a Term the Labeler Doesn't Control



Self term

Model's own sharpened prediction supplies a correction the labeler can't touch.



Schedule

Blend weak and self terms on a ramp. The self term is worthless at step zero.



Reliability weight

Weight each weak target by cross-source agreement, computed with no labels.



Gating

Discard targets that carry no usable signal rather than training on them weakly.



Performance Gap Recovered

PGR = 0%

student reproduces the labeler exactly

PGR = 80%

confidence loss on NLP tasks (Burns et al.)

PGR = 100%

weak labels as good as gold (unrealistic)

- **$PGR = (\text{student} - \text{weak}) / (\text{ceiling} - \text{weak})$**
- Requires a gold set to compute the denominator
- nDCG can rise while PGR falls, so always track both



02

PART 02

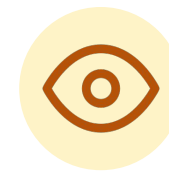
Mapping This Onto Retrieval



Weak Supervisors Already in Your Stack

Source	Coverage	Marginal cost
BM25 score	every candidate	already logged
Bi-encoder score	every candidate	already logged
Small cross-encoder	top-k only	already logged
Debiased CTR	head queries only	needs propensities





Cross-Encoders Already Know What “Relevant” Means

- Cross-encoders learn relevance during pretraining before fine-tuning
- Untrained, they still beat BM25 on several benchmark datasets
- Fine-tuning mostly teaches vocabulary, not relevance judgment itself
- So weak labels just need to point at a skill the model already has





Use Ranks Instead of Raw Weak Scores

- Raw BM25 scores not comparable across queries or across sources
- But order *within* a query still holds up
- Train on order, not the raw numbers

Query A		Query B	
Doc 1	score 14.2	Doc 4	score 31.5
Doc 2	score 11.8	Doc 5	score 28.9
Doc 3	score 9.4	Doc 6	score 22.1

14.2 vs 31.5 not comparable. Not the same units nor distribution.

But within each query, order survives:

Doc 1 > Doc 2 > Doc 3
Doc 4 > Doc 5 > Doc 6

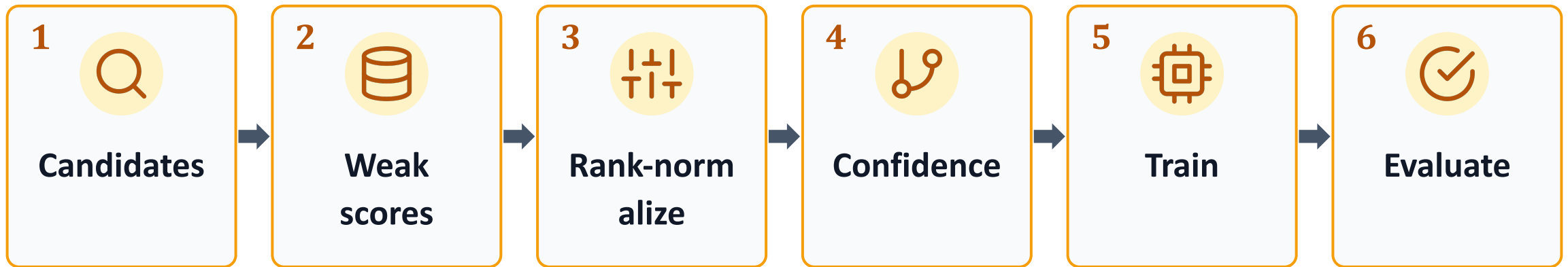
03

PART 03

The Pipeline



Six-Stage Pipeline



Stage 1: Don't Let the Labeler Pick the Candidates

1. Candidates

2. Weak scores

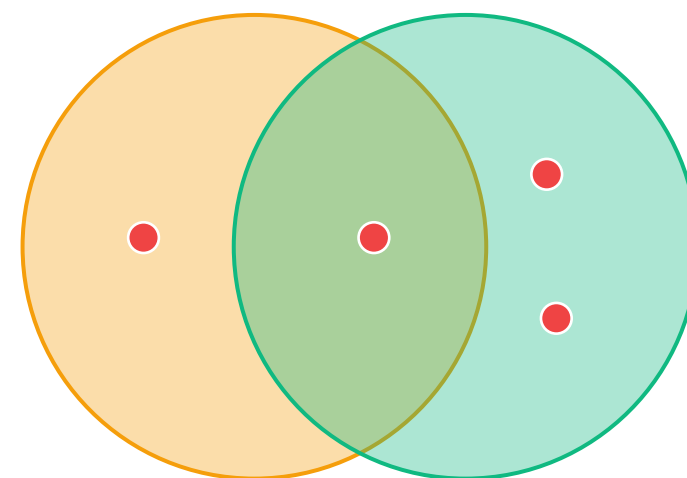
3. Rank-normalize

4. Confidence

5. Train

6. Evaluate

- BM25 pool + BM25 labels = a closed system
- The student never sees documents BM25 missed
- Union: BM25 top-k, dense top-k, random negatives
- **Recall@100 on gold bounds every downstream metric**
- Measure it before training anything



BM25

Dense

● *gold-relevant docs BM25 misses entirely*

Stage 2: Combine Sources With Decorrelated Errors

1. Candidates

2. Weak scores

3. Rank-normalize

4. Confidence

5. Train

6. Evaluate

- BM25 fails on paraphrase; dense retrieval fails on rare identifiers
- Disagreement between sources is a free uncertainty estimate
- Reranker only evaluated top-k; score anything outside it as “unknown” rather than “low”
- Debias clicks before inclusion, since even a crude position prior beats raw CTR
- Log per-source scores individually, not just the fused result



Stage 3: Rank-Normalize Per Query, Per Source

1 Rank

Convert raw scores into order: 1st, 2nd, 3rd...

2 Weight

Favor top ranks, reciprocal-rank style

3 Soften

Temperature-scaled distribution becomes target

EXAMPLE WITH THREE DOCUMENTS

Doc	Score	Rank	Target
A	14.2	1	0.42
B	11.8	2	0.35
C	9.4	3	0.23

Sharp raw gaps (14.2 vs. 9.4) compress into close, comparable targets (0.42 vs. 0.23)

Too little softening = right back to hard labels

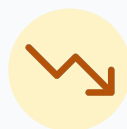


Stage 4: Three Label-Free Reliability Signals



Agreement

Fraction of sources placing this document in the top-k. Split sources mean the target is close to arbitrary.



Margin

Score gap to the next-ranked document within a source. A thousandth-of-a-point gap is noise caused by a floating-point.



List Entropy

A flat score distribution over the candidate list means that source isn't discriminating between candidates for this query.



Stage 5: Gating & Blending

1

Gate

Drop pairs with no real signal to give



2

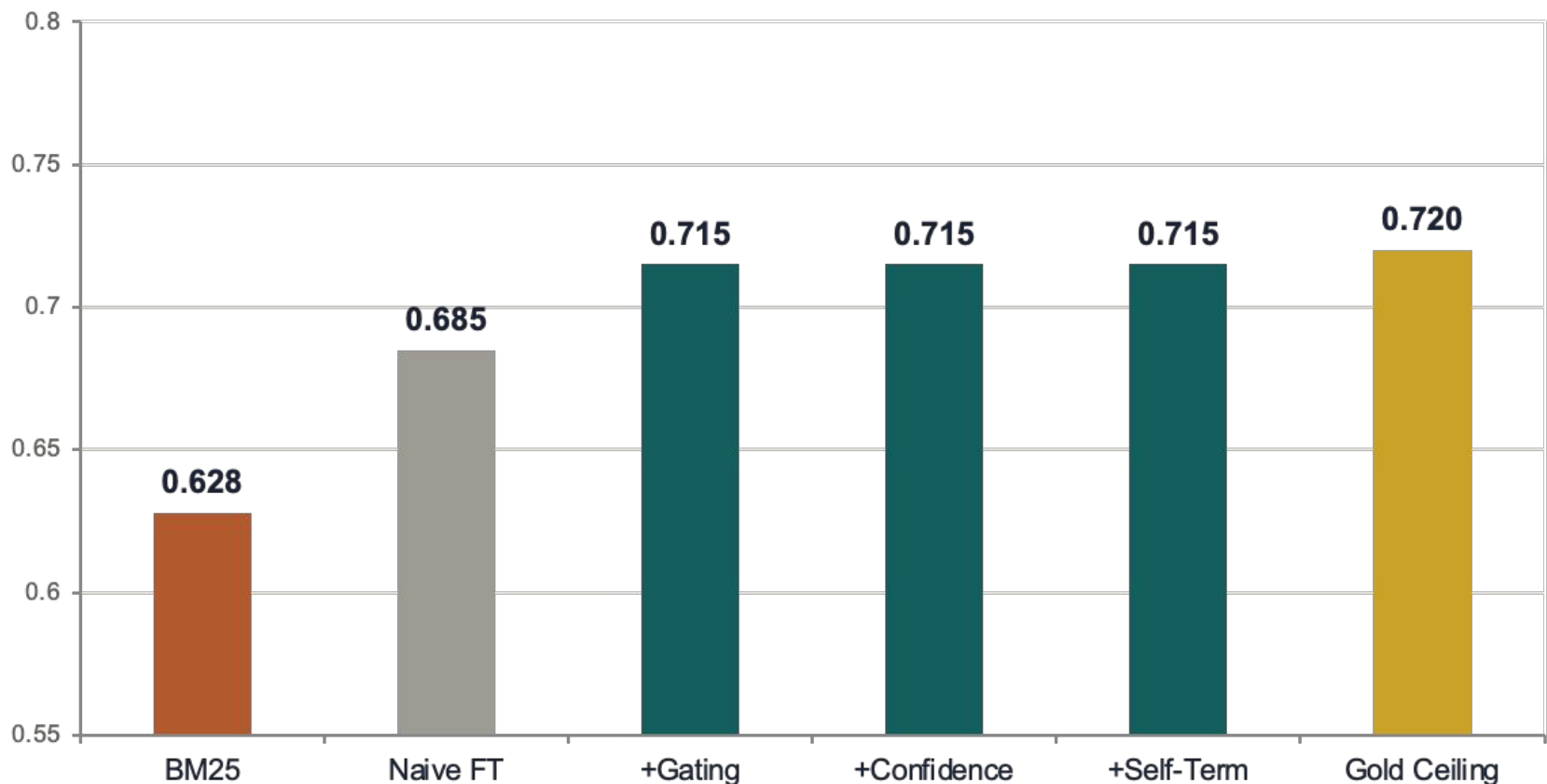
Blend

Weak label + model's own belief, on a schedule

Zero gradient beats a gradient toward noise.



Incremental Results



PGR BY STAGE

Naive FT	0.621
+ Gating	0.943
+ Confidence	0.940
+ Self-Term	0.940

Disagreement precision

0.289 → 0.417 → 0.405 → 0.405 (gold: 0.421)

ECE

0.066 → 0.065 → 0.068 → 0.068 (gold: 0.054)

Gating did the work: 0.62 → 0.94 PGR from adding the confidence gate alone.
 Weighting and the self-training term added nothing further on top; flat at 0.94, within noise on n=30.

04

PART 04

Instrumentation and Failure Modes



Four Metrics to Log Every Evaluation Step

nDCG

gold set: the only
unbiased signal

Agree@10

with labeler; should
decline, then plateau

Disagr. Prec.

student win rate when
they differ

ECE

whether confidence
carries information

- **Disagreement precision below 0.5 means back off alpha.**
- Neither alpha failure mode below is visible in training loss alone.





Failure 1: Agreement Never Declines



Symptom

Agreement@10 flat near 0.95.
Gold nDCG improves slightly,
then plateaus.



Cause

Alpha ceiling too low, or the
ramp starts too late in training.



Fix

Raise the alpha ceiling and start
the ramp earlier. Benign; you've
reproduced naive fine-tuning.





Failure 2: Self-Reinforcing Errors



Symptom

Agreement and gold nDCG both collapse sharply, shortly after alpha exceeds ~ 0.5 .



Cause

Self term became active before the model learned the task which locked onto arbitrary early predictions.



Fix

Longer warmup, lower alpha ceiling, higher tau. This does not recover mid-run, so restart.





Failure 3: Correlated Weak Sources



Symptom

Confidence estimator drives weight to maximum on exactly the examples where all sources share a blind spot.



Cause

BM25, tf-idf, and title match measure nearly the same thing, so counting them as three independent votes inflates one shared bias.



Fix

Compute Kendall's tau between sources. Collapse or drop anything correlated above ~ 0.9 .





Failure 4: Pool Recall Ceiling



Symptom

Strong offline gains, zero online movement. Survives code review and offline eval.



Cause

Candidate pool came from the weak labeler, so the evaluation pool shares the same blindspots.



Fix

Measure Recall@100 against gold, separately from ranking metrics. Report it permanently.



05

PART 05

When to Use Something Else



Choosing an Approach

Condition	Approach
Weak signals cheap, gap > 5 nDCG points	Weak-to-strong
Teacher more capable than student	Distillation
Relevance defined by business rules	Human labels
Sources share a systematic bias	Debias first
Regulated or safety-critical ranking	Human labels





Elicitation Requires Something to Elicit



Works

“does this passage answer this query?”

General relevance judgment sits in the pretraining distribution; weak labels only have to point at it.



Fails

“does this clause violate our escalation policy?”

That definition only lives in your internal documents, so there is nothing latent to extract.

- No loss function substitutes for knowledge the model does not have
- **Test zero-shot on 20 of your own queries before committing to this approach at all**



Sizing the Gold Evaluation Set

300–500

queries, judged to depth 10

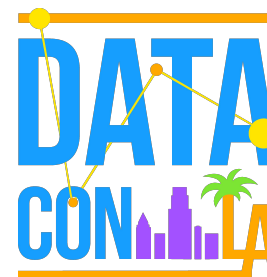
$\sigma \approx 0.15$

typical per-query nDCG@10
variance

~ 0.02

detectable nDCG difference at that
sample size

- Below roughly 200 queries, seed noise dominates method differences
- Stratify across head, torso, and tail query frequency
- Roughly one engineer-week or a small annotation spend; run the power calculation on your own variance



06

PART 06

Recap





What to Take Away

- Cross-entropy against weak labels is minimized by imitation, errors included
- A scheduled self-consistency term breaks that ceiling without discarding weak signal
- Rank-normalize scores, gate low-margin pairs, weight targets by source agreement
- Track disagreement precision **and** nDCG to catches failures not captured in training loss
- Buy 300–500 judged queries strictly for evaluation
- Check pool recall before tuning the reranker, as it bounds everything downstream



Code, Evaluation Harness, and Reference Configs

github.com/devyn-miller/datacon-la-2026-w2sg

Burns et al. 2023, arXiv:2312.09390 · Dehghani et al. 2017, arXiv:1704.08803

Joachims et al. 2017, WSDM (unbiased LTR) · Thakur et al. 2021, BEIR



Thank you for your time! If you'd like to...

Submit feedback:



Look at the code:



**Connect on
Linkedin:**

