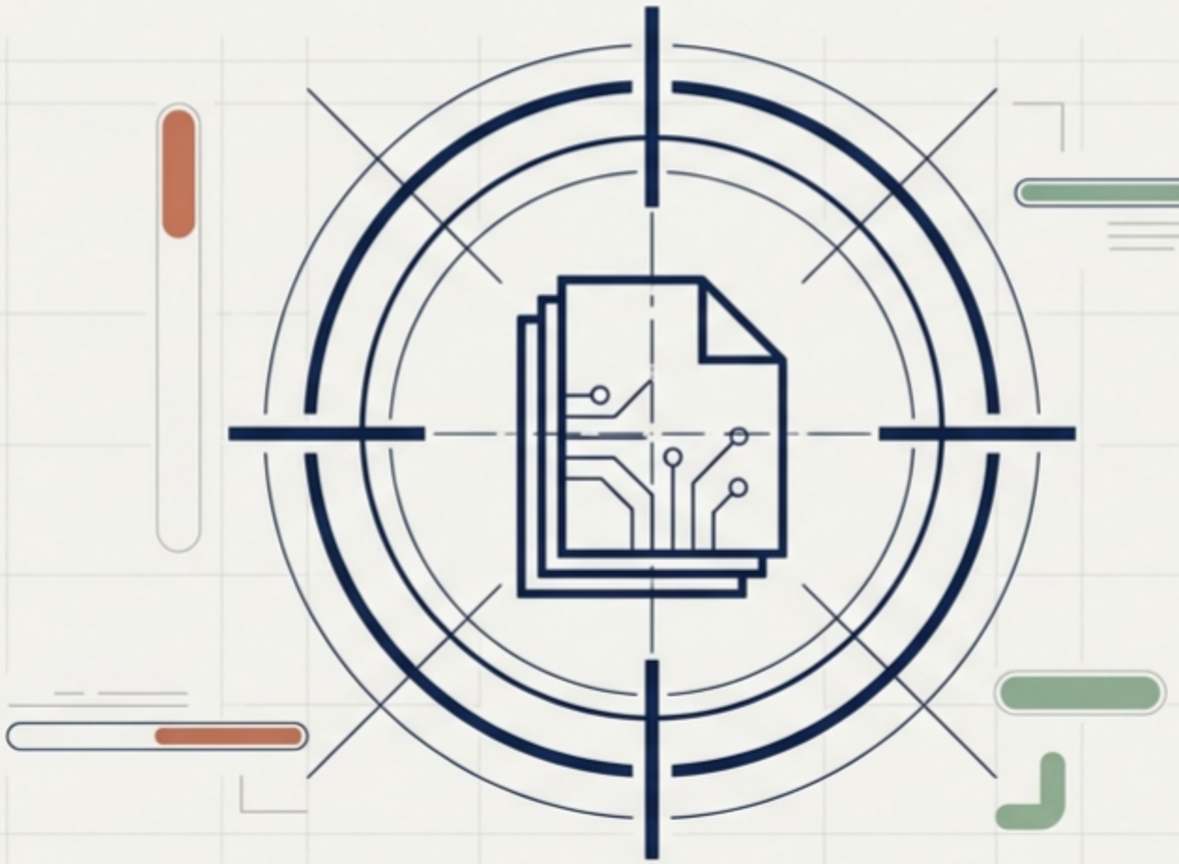
CALIBRATING SEARCH

Turning Vibes into Telemetry with IR Metrics



SYSTEM: STANDBY | MODE: DIAGNOSTIC

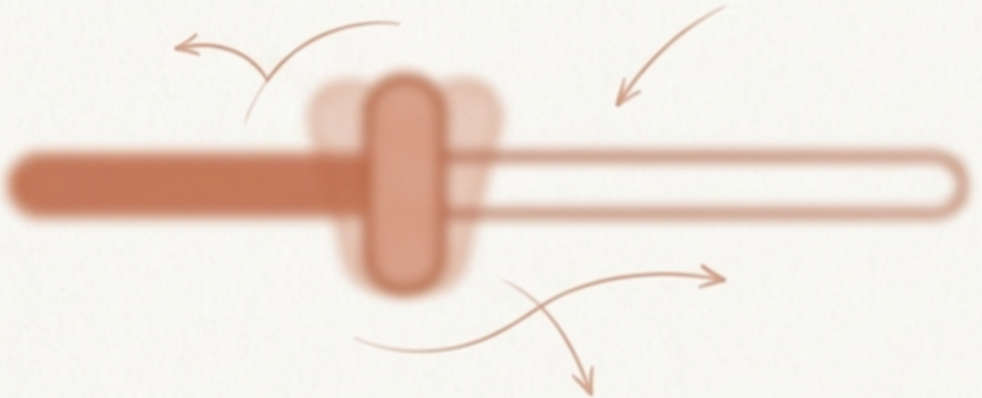
//DATA.LOG: 0X2F4A1E8 //TEMP: 298K //AXIS: Y-7 //STREAM: ACTIVE //PROTOCOL: V4 //NODE: 14

//DATA.LOG: 0X2F4A1E8 //TEMP: 298K //AXIS: Y-7 //STREAM: ACTIVE //PROTOCOL: V4 //NODE: 14

//DATA.LOG: 0X2F4A1E8 //TEMP: 298K //AXIS: Y-7 //STREAM: ACTIVE //PROTOCOL: V4 //NODE: 14

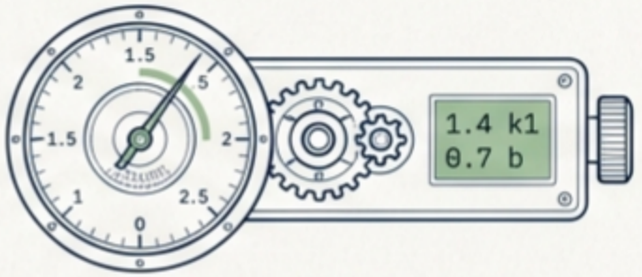
You Cannot Improve What You Don’t Measure

The Problem with Guessing

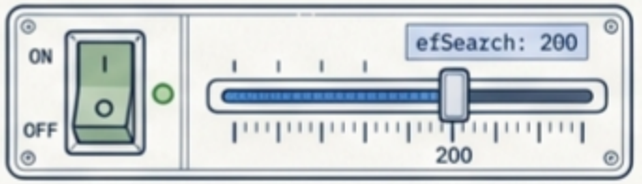


Relying on “this feels better” breaks down at scale.

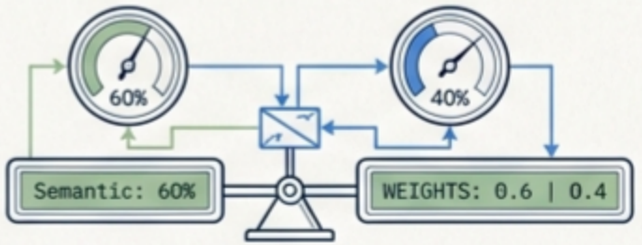
The Necessity of Measurement



BM25: Tuning k1 and b



HNSW: Setting efSearch



Hybrid: Balancing semantic vs. keyword weights

Every tuning choice must be decided by measuring against fixed targets.

The Anchor: Relevance Judgments

Step 1: Raw Query

Step 2: Human Evaluation

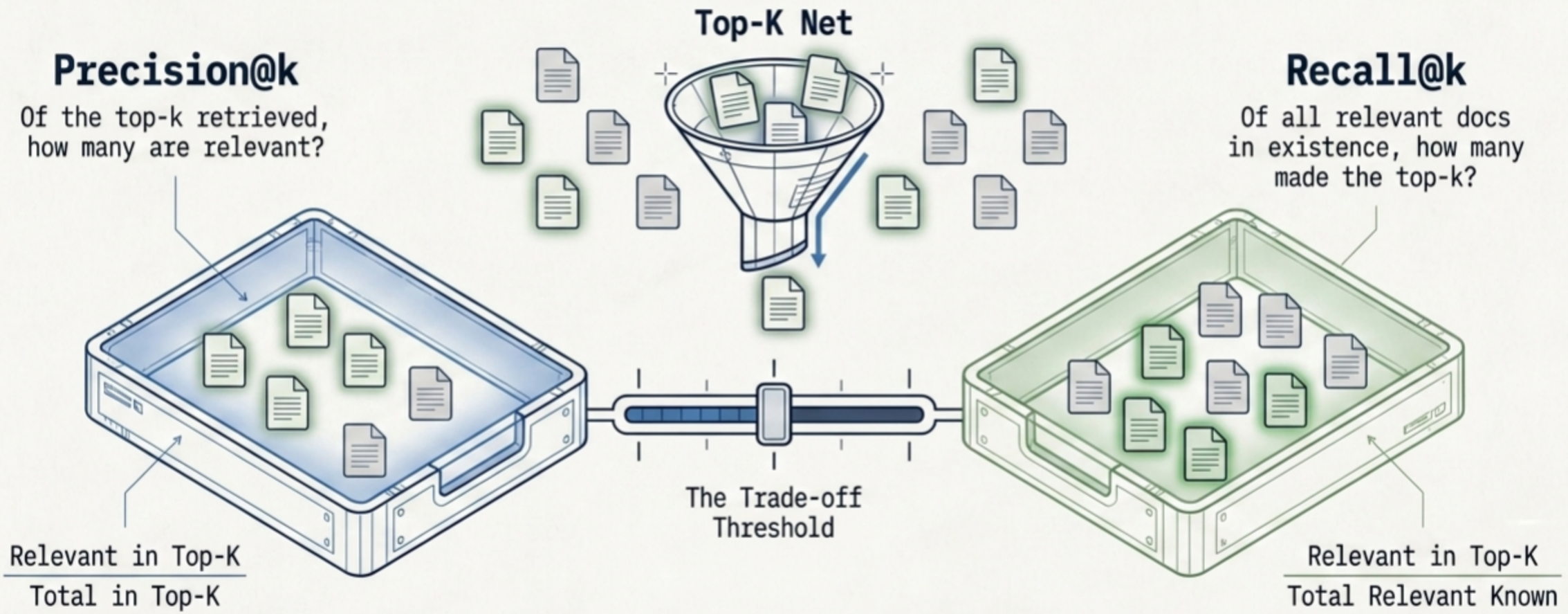
Step 3: The Anchor File

Human experts review retrieved documents.

Ground Truth = Queries securely paired with documents that a human has judged as relevant.

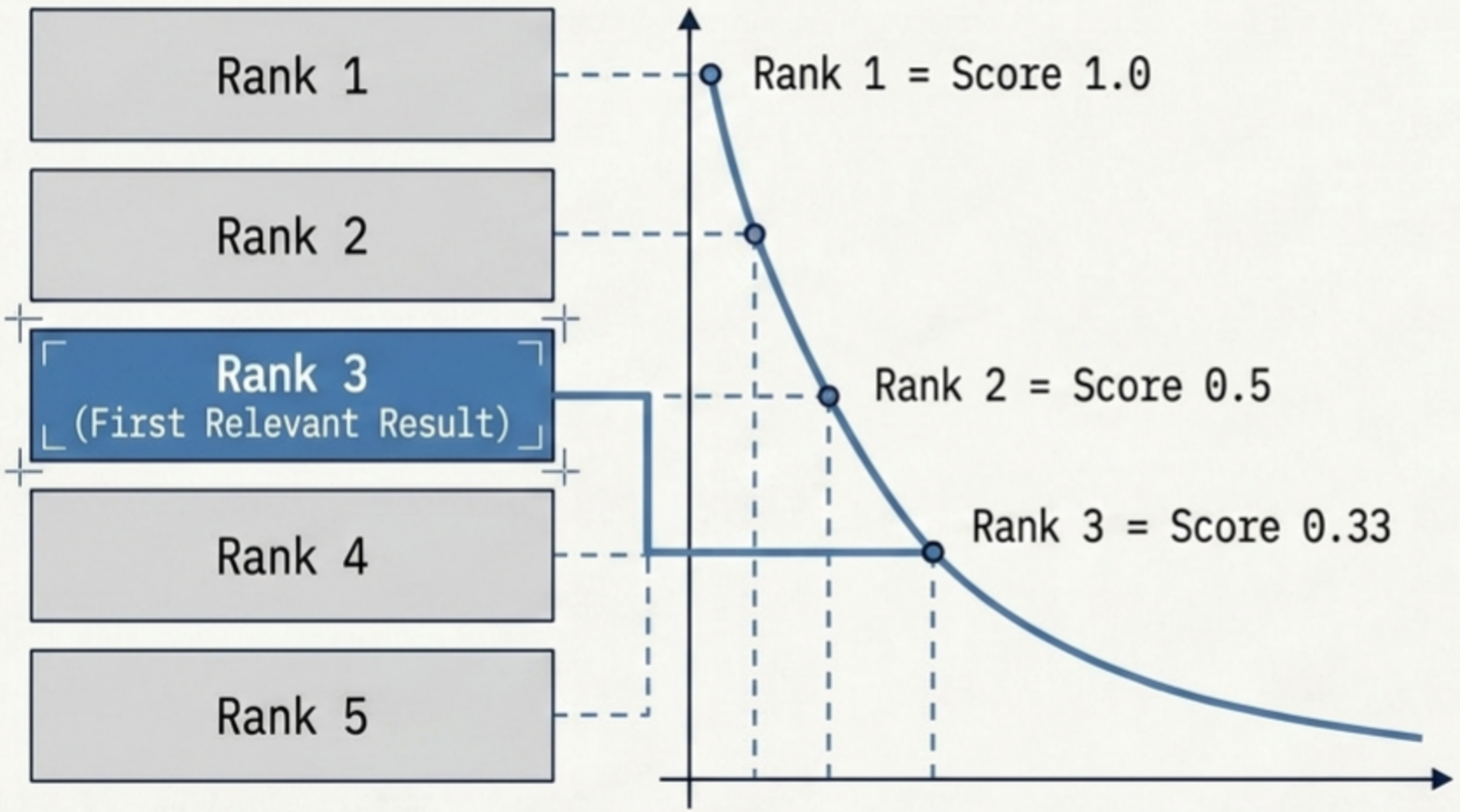
This labeled set acts as the immovable anchor for all subsequent mathematical evaluation.

The Baseline Constraints: Precision@k vs. Recall@k



Expanding k increases Recall but usually degrades Precision. They are inherently traded off against each other.

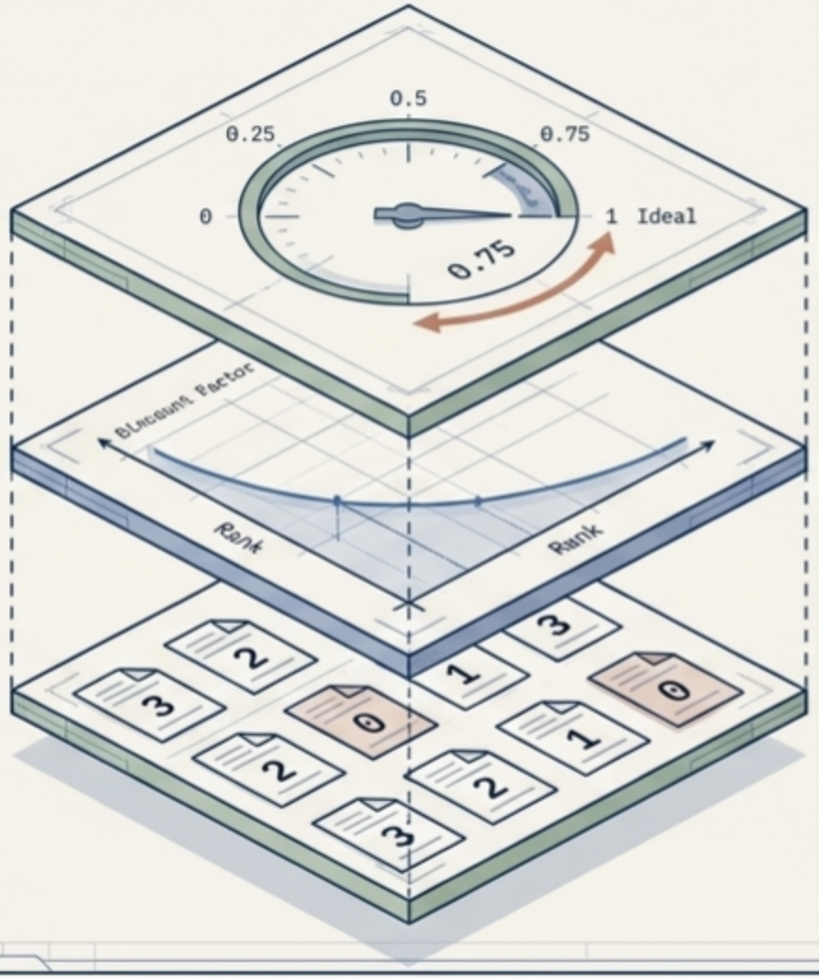
The Top-Hit Metric: MRR (Mean Reciprocal Rank)



Average of 1 / (rank of the first relevant result) across multiple queries.

When to use?
Best when only the single top answer matters (e.g., 'I'm Feeling Lucky' or known-item lookups).

The Graded List Metric: nDCG



1. Graded Relevance (CG) - Documents aren't just "good/bad"; they are scored on a spectrum (e.g., 0 to 3).

2. Positional Discount (D) - Rewards putting the best results near the top. Applies a logarithmic penalty to lower ranks.

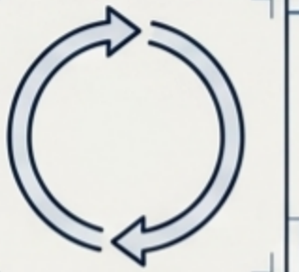
3. Normalization (n) - Divides by the "Ideal" sequence so scores always fall within [0,1], making different queries perfectly comparable.

When to use? The go-to metric when order matters across the entire list.

The Metric Selection Toolkit

Metric	Best For...	Blind Spot	Output Scale
`Precision@k`	Measuring “noise” in the top results	Ignores relevant docs missed entirely	`% of Top-K`
`Recall@k`	Ensuring no good answers are left behind	Ignores how far down the list they are	`% of All Relevant`
`MRR`	Factoid search, single-answer lookups	Ignores the 2nd, 3rd, and 4th results	`0 to 1 (1/x)`
`nDCG`	E-commerce, graded relevance lists	Complex to calculate and explain	`Normalized [0,1]`

Evaluation Environments: Offline vs. Online

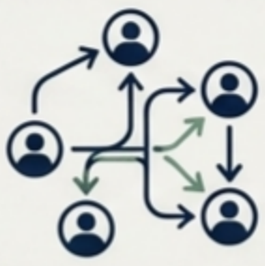


Offline Evaluation
(The Lab)

Data: Fixed, labeled sets (Relevance Judgments).

Speed: Instant feedback. Run algorithms against the same ground truth thousands of times.

Limitation: Static. Doesn't account for shifting user intent.



Online Evaluation
(The Real World)

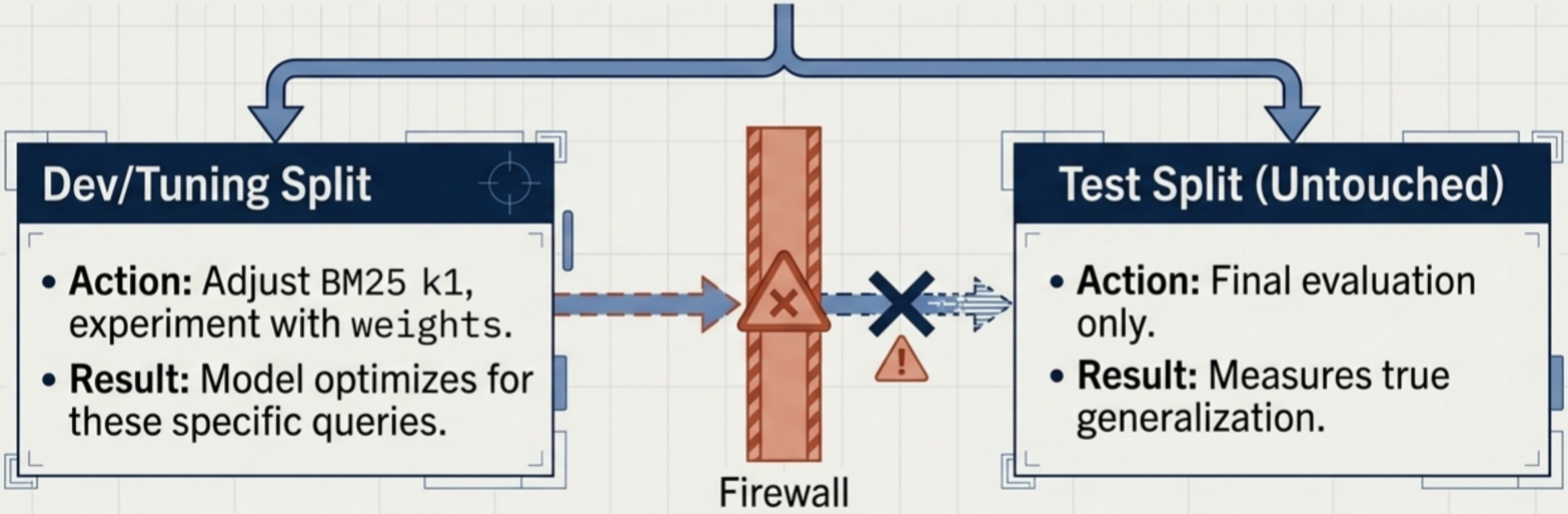
Data: A/B Testing real user behavior (Clicks, conversions).

Speed: Slower. Requires traffic and time to reach statistical significance.

Strength: The ultimate ground truth of user satisfaction.

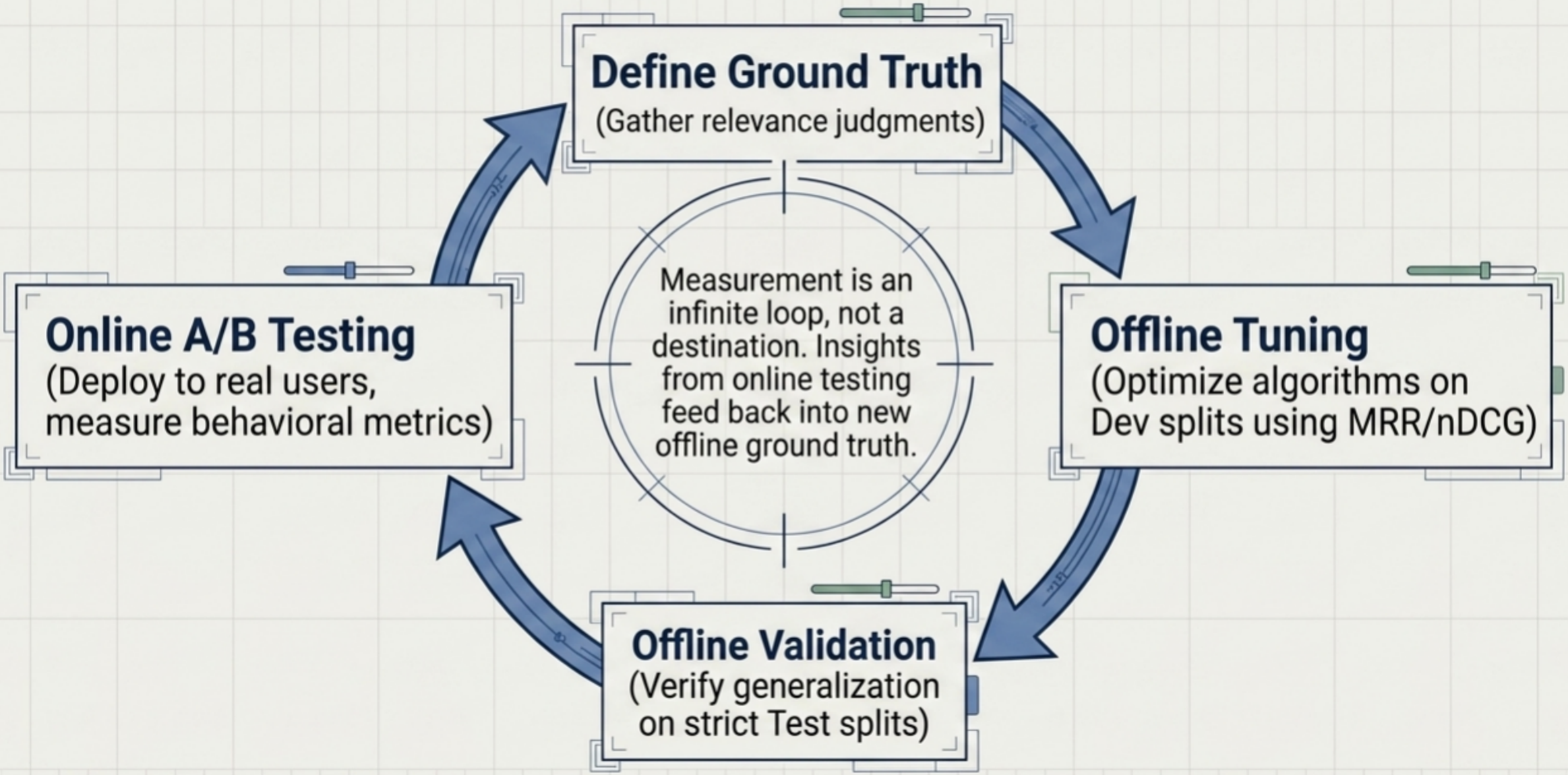
Critical Pitfall: Data Contamination

Never tune hyperparameters on the same queries you report on.



The Danger of Overfitting: If you tune on your test set, settings look perfect in the lab but fail entirely in the real world. Also, avoid **tiny test sets**—they are statistically noisy.

The Search Evaluation Lifecycle



Diagnostic Readout: Check Your Understanding

SYSTEM PROMPTS >	DIAGNOSTIC ANSWERS >
Q: Precision@k?	**A: Of the results shown, how many are good?**
Q: Recall@k?	**A: Of all good results, how many were found?**
Q: MRR?	**A: How high was the first hit?**
Q: nDCG?	**A: Are the absolute best results near the top?**

Why normalize nDCG?

To make queries comparable; divides by ideal DCG so scores fall in [0,1].

Danger of tuning on test set? ⚠

Overfitting. Settings won't generalize. Always maintain a Dev/Test firewall.