A/B Test Report:Similarity-aware Difficulty Scoring

David Zhu* Catharine Li†

August 2025

Abstract

This report evaluates a post-processing rule that lowers the difficulty score for concepts whose names closely match previously seen cards (found in the `data/` folder). We compare the baseline scoring (A) with the similarity-aware adjustment (B) across multiple thresholds and penalty strengths.

1 Objective

Assess whether reducing difficulty for repeated or near-duplicate concept names (based on string similarity) behaves as intended while leaving new concepts unaffected.

2 Methodology

Baseline (A). The base difficulty was fixed at 4 for testing.
Variant (B). Apply:

- `similarity_threshold` $\in \{0.75, 0.82, 0.90\}$
- `lower_by` $\in \{1, 2\}$
- If a prior card name in `data/` exceeds the threshold similarity with the current concept name, decrease difficulty by `lower_by` (floor at 1).

Concepts tested: *Code Documentation*, *Parameter Efficient Fine-Tuning*, *Completely New Topic*, *Testing Documentation*.

3 Results

Baseline (A) difficulty for all rows: 4. The table reports the adjusted difficulty (B).

Summary. Repeated concepts (e.g., *Code Documentation*, *Testing Documentation*) drop by the configured penalty when similarity exceeds the threshold. New concepts (*Completely New Topic*) stay at the baseline. At the strictest threshold (0.90), *Parameter Efficient Fine-Tuning* no longer triggers a reduction, indicating the threshold is working as a precision control.

*rz2718@nyu.edu
†cl5429@nyu.edu

4 Discussion & Recommendation

The mechanism behaves as designed. For production, we recommend:

- **similarity_threshold = 0.82:** balances catching legitimate repeats while avoiding over-matching.
- **lower_by = 1:** conservative reduction that avoids overly deflating difficulty on borderline matches.

Future work: run on a larger, real set of user-generated cards; log hit rate (how often reductions occur) and collect qualitative feedback.

Appendix A: Repro Snippet

Minimal driver used to generate Table **??** (base difficulty set to 4):

```
for concept in ["Machine Learning",
                "Parameter Efficient Fine-Tuning",
                "Dimensionality Reduction",
                "TRetrieval-Augmented Generation"
                "LangChain Expression Language",
                "Vector Database"]:
    for threshold in [0.75, 0.82, 0.90]:
        for lower_by in [1, 2]:
            adjusted = adjust_difficulty_based_on_history(
                concept_name=concept,
                base_difficulty=4,
                data_dir="data",
                similarity_threshold=threshold,
                lower_by=lower_by
            )
            print(concept, threshold, lower_by, adjusted)
```

Concept	Threshold	Lower By	Adjusted Difficulty
Machine Learning	0.75	1	3
Machine Learning	0.75	2	2
Machine Learning	0.82	1	3
Machine Learning	0.82	2	2
Machine Learning	0.90	1	3
Machine Learning	0.90	2	2
Parameter Efficient Fine-Tuning	0.75	1	3
Parameter Efficient Fine-Tuning	0.75	2	2
Parameter Efficient Fine-Tuning	0.82	1	3
Parameter Efficient Fine-Tuning	0.82	2	2
Parameter Efficient Fine-Tuning	0.90	1	4
Parameter Efficient Fine-Tuning	0.90	2	4
Dimensionality Reduction	0.75	1	3
Dimensionality Reduction	0.75	2	2
Dimensionality Reduction	0.82	1	3
Dimensionality Reduction	0.82	2	2
Dimensionality Reduction	0.90	1	3
Dimensionality Reduction	0.90	2	2
TRetrieval-Augmented Generation	0.75	1	3
TRetrieval-Augmented Generation	0.75	2	2
TRetrieval-Augmented Generation	0.82	1	3
TRetrieval-Augmented Generation	0.82	2	2
TRetrieval-Augmented Generation	0.90	1	4
TRetrieval-Augmented Generation	0.90	2	4
LangChain Expression Language	0.75	1	3
LangChain Expression Language	0.75	2	2
LangChain Expression Language	0.82	1	3
LangChain Expression Language	0.82	2	2
LangChain Expression Language	0.90	1	4
LangChain Expression Language	0.90	2	4
Vector Database	0.75	1	3
Vector Database	0.75	2	2
Vector Database	0.82	1	3
Vector Database	0.82	2	2
Vector Database	0.90	1	3
Vector Database	0.90	2	2

Table 1: Adjusted difficulty levels for various ML/DL concepts across thresholds and lower_by values.