Cognitive-AI Approaches to Adaptive Scheduling in Braille Memory Retention Using Hybrid SM-2 Models

RESEARCH PROPOSAL

PRESENTED BY:

Jonathan Leewin

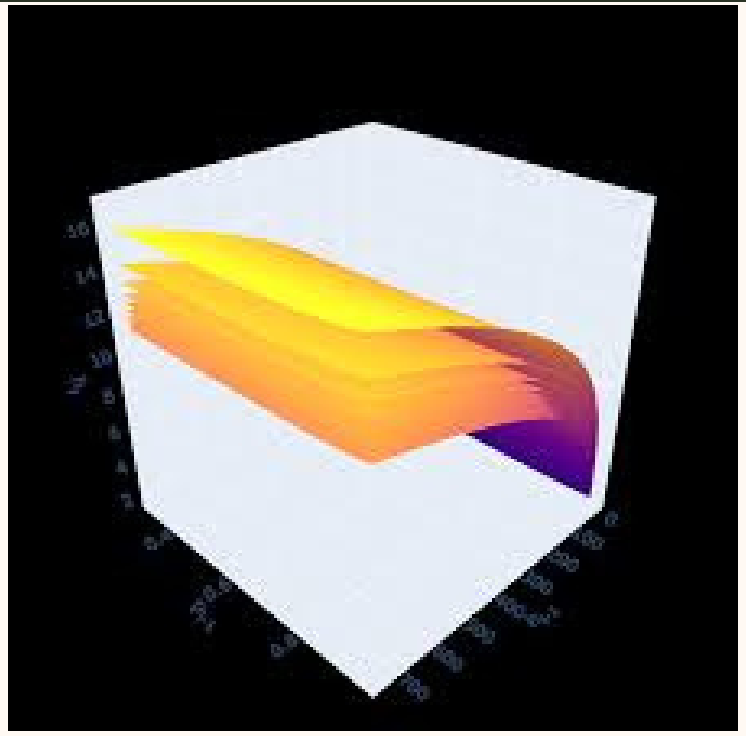
COMP6065001

INTRODUCTION

Only about 10-25 percent of legally blind individuals in the U.S. are fluent in Braille, though those who are fluent tend to have far better employment and literacy outcomes than those who are not. (The Stanford Daily) Spaced repetition reliably improves long-term retention across domains, reducing forgetting and enhancing learning efficiency. (PMC) Current algorithms like SM-2 offer transparency but limited individual adaptation while machine learning based schedulers adapt well but are often opaque. The work proposed here combines the strengths of SM-2 and ML to schedule Braille review using IoT-enabled tactile flashcards, optimizing recall curves, time to high recall, forgetting penalties, review timing, and daily study burden.

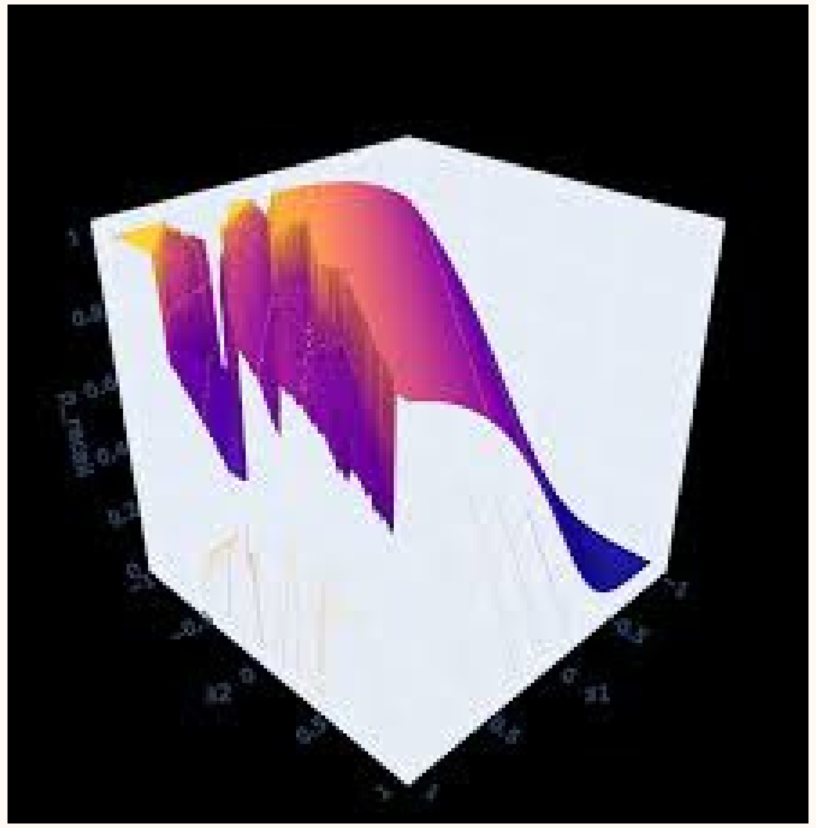
BACKGROUND

Braille literacy functions as a pattern recognition system where tactile dot combinations correspond to letters and symbols, similar to visual flashcard learning through pattern-based memory formation. Traditional SM-2 algorithms demonstrate rigid limitations that cannot adapt to individualized learning patterns, while pure machine learning approaches suffer from computational complexity, overfitting tendencies, and black-box operations lacking interpretability. These shortcomings create an urgent need for adaptive algorithmic solutions that balance interpretability with personalization for effective pattern-based learning systems.



SM-2

ML-based approach



PROBLEM

Spaced repetition scheduling algorithms face trade-offs that limit their effectiveness in educational applications. SM-2 offers interpretability but remains static and cannot adapt to individual learning patterns. FSRS provides predictive capabilities but is constrained by fixed parametric forms that limit flexibility. Machine learning approaches deliver adaptability but operate as opaque systems without clear decision rationale.

SM-2

- Transparent & easy to understand
- Static, not adaptive
- Limited personalization

ML - Based

- Opaque decisions
- Lack of transparency
- Risk of overfitting

OBJECTIVES

1. Develop baseline (SM-2), ML, and hybrid scheduling methods
2. Compare scheduling policies using SSP-MMC-Plus metrics
3. Apply to Braille literacy with IoT tactile flashcards

LITERARY REVIEW

- Jankowski, J. (2022, November 2). Application of a computer to improve the results obtained in working with the SuperMemo method - SuperMemo. SuperMemo. https://www.supermemo.com/en/blog/application-of-a-computer-to-improve-the-results-obtained-in-working-with-the-supermemo-method
- Su, J., Ye, J., Nie, L., Cao, Y., & Chen, Y. (2023). Optimizing spaced repetition schedule by capturing the dynamics of memory. IEEE Transactions on Knowledge and Data Engineering, 35(10), 10085–10097. https://doi.org/10.1109/tkde.2023.3251721
- Ye, J., Su, J., & Cao, Y. (2022). A stochastic shortest path algorithm for optimizing spaced repetition scheduling. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 4381–4390. https://doi.org/10.1145/3534678.3539081

SCHEDULING ALGORITHMS

SM-2

- Each review outcome (quality score qq) updates ease factor (EF) and repetition count.
- Intervals grow approximately exponentially with EF.
- Limitation: uniform treatment of items and learners.

$$EF' = EF + (0.1 - (5-q)(0.08 + (5-q) \times 0.02))$$

FSRS

- SS = stability (half-life proxy).
- On recall success, stability increases; on failure, it decreases.
- Parameters optimized with gradient descent.
- Recall Probability modeled as the one below:

$$p(\Delta t) = e^{-\Delta t/S}$$

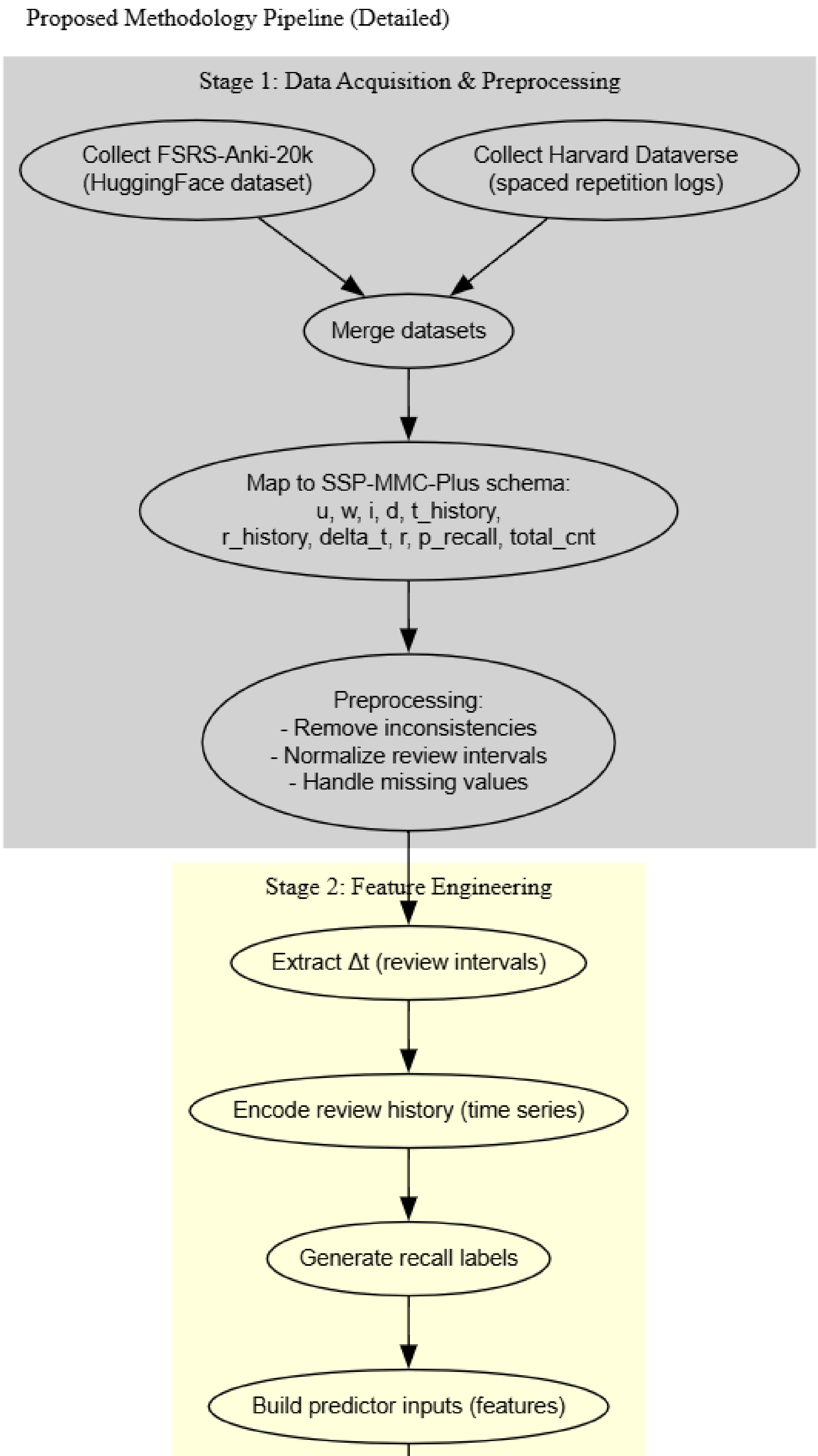
Hybrid SM-2 + AI (Proposed)

- Calculate the next review interval using the standard SM-2 algorithm (I _ sm2)
- Use a machine learning model to estimate the learner-specific memory half-life and derive a predicted interval (i _ ML)
- Combine the SM-2 and ML intervals using a blending factor β shown below
- Adjust the blending factor based on the model confidence or the learner's stage to optimize interval prediction.

$$I_{next} = (1-\beta)I_{SM2} + \beta I_{ML}$$

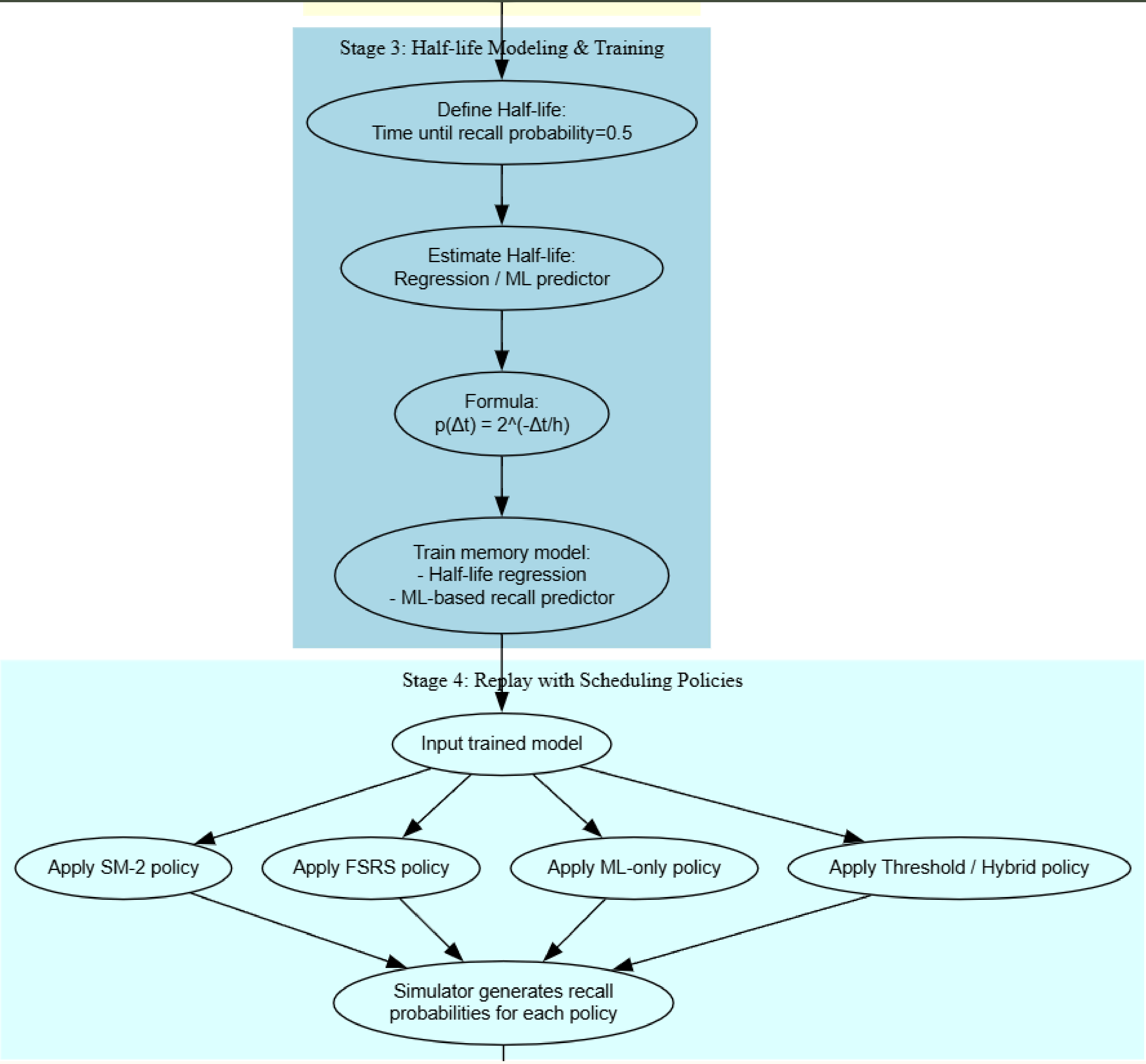
METHODOLOGY

Pipeline



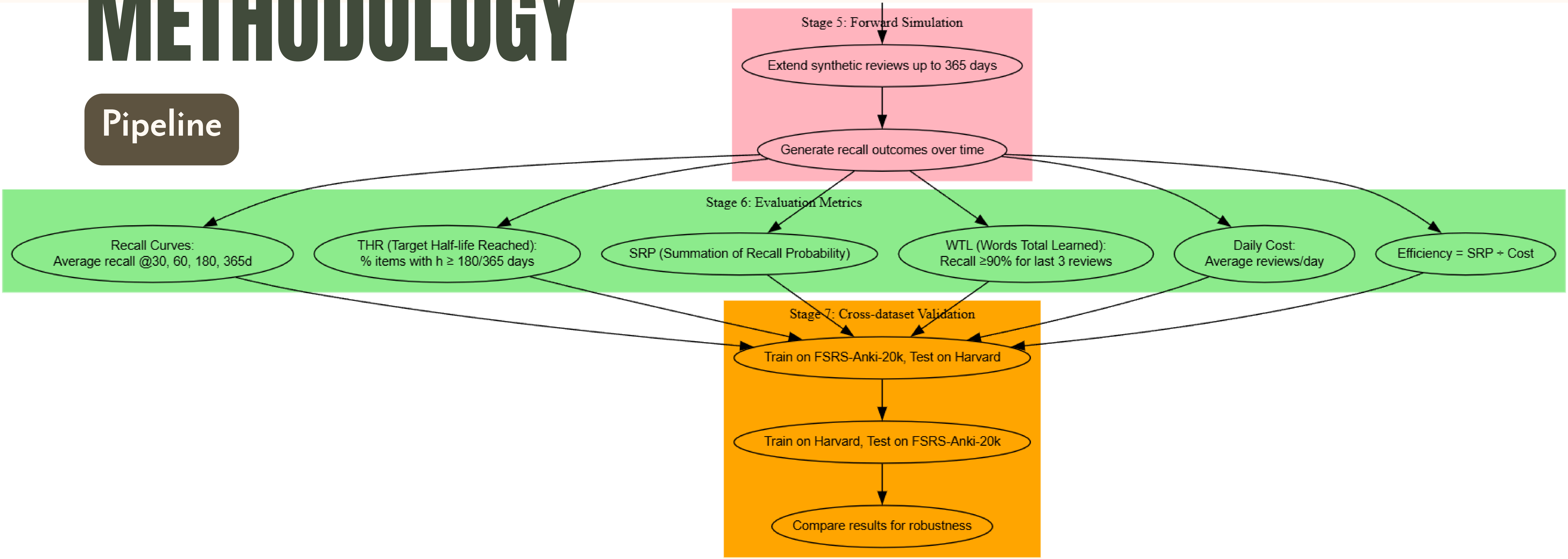
METHODOLOGY

Pipeline



METHODOLOGY

Pipeline



METHODOLOGY 1.0

Datasets

- FSRS-Anki-20k (HuggingFace): Review histories of 20k Anki users, millions of entries.
- Harvard Dataverse spaced repetition dataset: Additional learner logs.
- Both mapped to SSP-MMC-Plus schema:
- u, w, i, d, t_history, r_history, delta_t, r, p_recall, total_cnt.

METHODOLOGY 2.0

Why Half-life?

- Definition: Time until recall probability falls to 0.5.
- Importance:
  - Captures durability of memory beyond single recall accuracy.
  - Directly tied to scheduling: review before half-life ends.
  - Provides fair, comparable metric across algorithms.

Formula:

The recall probability after a time interval Δt is given by:

$$p(\Delta t) = 2^{-\Delta t/h}$$

where h is the half-life of the memory (time until recall probability falls to 0.5).

Example:

If the recall probability after 20 days is 0.45, we can solve for h:

$$0.45 = 2^{-20/h} \implies -\frac{20}{h} = \log_2 0.45 \implies h \approx 14.8 \text{ days}$$

This means the memory's half-life is approximately 14.8 days.

METHODOLOGY 3.0

Evaluation Metrics

- Recall Curves: average recall probability at 30, 60, 180, 365 days.
- THR (Target Half-life Reached): % of items with half-life ≥ 180 or 365 days.
- SRP (Summation of Recall Probability):
- SRP=Σi=1Nπ
- WTL (Words Total Learned): items with recall ≥90% for last three reviews.
- Daily Cost: average reviews/day. Efficiency = SRP ÷ Cost

METHODOLOGY 4.0

Evaluation Procedure

- Fit Memory Model
  - Train half-life regression or ML predictor on datasets.
  - Produces recall probability estimates.
- Replay with Policies
  - Apply SM-2, FSRS, ML-only, Threshold, and Hybrid policies in simulator.
  - Each policy selects intervals.
  - Simulator generates recall outcomes probabilistically.
- Forward Simulation
  - Extend reviews up to 365 days.
  - Generate synthetic recall for long-term outcomes.
- Compute Metrics
  - At checkpoints (30, 60, 180, 365 days), calculate Recall Curves, THR, SRP, WTL, Daily Cost.
- Cross-dataset Validation
  - Train on FSRS-Anki-20k, test on Harvard.
  - Reverse for robustness.

EXPECTED CONTRIBUTIONS

- Comparative study of five scheduling algorithms: SM-2, FSRS, ML-only, Threshold, Hybrid.
- Development of a Hybrid SM-2+AI scheduler combining interpretability and adaptiveness.
- Demonstration of half-life as a key evaluation concept.
- Roadmap for Braille literacy applications via IoT tactile flashcards.

INPUT

- good scope
- there is possibility the ml predictor does not work because according to ken, in a practical sense it dosnt simulate a human brain 1:1
-

ROADMAP

Phase 1

Implement SM-2, FSRS

Phase 2

Train ML models

Phase 3

Develop Hybrid scheduler

Phase 4

Evaluate with SSP-MMC-Plus metrics