

Rule-based

① attempt weight

$$w_k = e^{-\lambda(N-k)}$$

↓
attempt index
decay rate
total attempt

$$w_k = e^{-\lambda(N-k)} + \frac{S_k}{\max score}$$

↓

② weighted score

$$\hat{S} = \frac{\sum_{k=1}^N w_k \cdot S_k}{\sum_{k=1}^N w_k}$$

③ mastery confidence per skill

$$\hat{S} \rightarrow \text{weighted score} \quad \theta_1 \rightarrow \text{threshold (based on the level)}$$

$\hat{S} - \theta_1$ ↗ num : ↑ expected mastery level
- num : ↓ expected mastery level

$$M_{skill, learner level} = \frac{1}{1 + e^{-\theta_2(\hat{S} - \theta_1)}}$$

④ condition normalization

mastery
engagement
time

$$C_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$$

⑤ rule activation

$$A_r = \prod_{i=1}^n C_i$$

rule importance

$$W_r \in [0, 1]$$

↳ from SFIA

- pre-requisite : close to 1
- exploration : lower

⑥ rule contribution score

$$S_r = A_r \cdot W_r$$

⑦ module score aggregation

$$S_m = \sum_{r \in R_m} S_r$$

total score
of the module

$$S_m = \sum_{r \in R_m} S_r$$

⑧ Final recommendation module

$$M^* = \arg \max S_m$$

Idea flow (single context):

For example, we have input:

- Learner: L1
- Target: SFIA skill = ARCH (level 3)
- Assessment attempts: 3 attempts

The raw results of those assessments:

Attempt k	Score S_k	Time
-------------	-------------	------

1	0.65	Long
2	0.75	Reasonable
3	0.80	Reasonable

From here, exponential attempt weighting, assuming we decide that $\lambda = 0.9$, we get:

Attempt	Calculation	Weight
1	$e^{-1(3-1)}$	0.13
2	$e^{-1(3-2)}$	0.37
3	$e^{-1(3-3)}$	1

Weighted score:

$$s = \frac{(0.13)(0.65) + (0.37)(0.75) + (1.00)(0.80)}{1.50} \approx 0.77$$

Next, for mastery confidence, assume that the threshold for level 4 is 0.75 ($\theta_4 = 0.75$) and $\beta_4 = 8$, then we get:

$$M_{u,s,l} = \frac{1}{1+e^{-8(0.77-0.75)}} \approx 0.54$$

On top of this, if we have high engagement (0.8) and acceptable learning speed (0.7), we have $C = \{0.54, 0.8, 0.7\}$, and based on the rule's importance:

- Skill gap rule ($M_{u,s,l} = 0.54$ and engagement = 0.8 | $W_1 = 0.7$):

$$s_1 = 0.54 \times 0.8 \times 0.7 \approx 0.30$$

- Prerequisite rule ($M_{u,s,l} = 0.54$ and mastery relevant = 0.8 | $W_2 = 0.85$):

$$s_2 = 0.54 \times 0.8 \times 0.85 \approx 0.37$$

- Engagement speed rule (engagement = 0.8 | learning speed = 0.7 | $W_1 = 0.1$):

$$s_3 = 0.8 \times 0.7 \times 0.1 \approx 0.06$$

Hence, $S_{module} = 0.30 + 0.37 + 0.06 = 0.73$.

We can compare the score is lower / standard / higher than other modules to check whether the module is recommended or not.

Based on this idea, the entire rule-based pipeline pseudocode should be like this:

```
INPUT:
Learner u
Assessment attempts per skill-level
Learning activity logs
Rule set R
Module set M

FOR each skill s and level l:
    Compute attempt weights:
         $w_k = \exp(-\lambda (N - k))$ 

    Compute weighted score:
         $s_{\hat{}} = \frac{\sum w_k * s_k}{\sum w_k}$ 

    Compute mastery confidence:
         $M[u, s, l] = \frac{1}{1 + \exp(-\beta_l (s_{\hat{}} - \theta_l))}$ 

FOR each rule r in R:
    Extract rule conditions:
         $C_i \leftarrow \text{normalize(relevant learner signals)}$ 

    Compute rule activation:
         $A_r = \text{product}(C_i)$ 

    Apply rule weight:
         $S_r = A_r * W_r$ 

FOR each module m in M:
    Aggregate rule scores:
         $S_m = \sum (S_r \text{ for all } r \text{ recommending } m)$ 

    Apply hard constraints:
        Remove m if prerequisites not met or time infeasible

SELECT:
 $m^* = \text{module with highest } S_m$ 

OUTPUT:
Recommended module list (ranked)
Explanation = contributing rules + signals
```