AQS: MoCA-Guided Adaptive Questioning with Learned Stopping for Efficient Cognitive Impairment Screening

Mingyu Li[1], Shuqiong Wu[2], Zhaojie Luo[1,3]
[1]School of Biological Science & Medical Engineering, Southeast University
[2]School of Engineering Science, Osaka University
[3]Shenzhen Loop Area Institute
lmy5492@163.com, luozhaojie@seu.edu.cn

Abstract

Fixed-item cognitive screening tools such as the Montreal Cognitive Assessment (MoCA) may ask redundant questions and increase patient burden. We propose AQS, an adaptive cognitive screening framework that selects questions and stops assessment according to an evolving cognitive state. Using ADNI-derived MoCA records, we construct an expanded question bank and an 8-dimensional state space, where free-form responses are mapped to discrete severity signals for online impairment updating. We compare handcrafted adaptive selection, fixed stopping, and reinforcement learning (RL) policies. Results show that combining a structured adaptive selector with an RL-based stopping policy yields the best performance–efficiency trade-off. Compared with the classical FisherInfo strategy, it improves Acc from 0.682 to 0.715 and AUC from 0.762 to 0.771, while reducing the average number of questions from 14.43 to 9.78. It also concentrates assessments in a more balanced question-count range and reduces excessive long-tail testing. Analysis further suggests that question selection and stopping are better treated as decoupled decisions, since naive joint RL leads to unstable optimization. These findings provide a simulation-grounded offline framework for efficient cognitive screening under data-scarce medical settings.

1 Introduction

Fixed-item screening instruments such as MoCA and MMSE are widely used for mild cognitive impairment (MCI) screening (Nasreddine et al., 2005; Folstein et al., 1975). Compared with imaging-based diagnosis, they are cheaper and easier to deploy, but they remain non-adaptive: all subjects answer the same items, redundant questions may increase patient burden, and administration still often requires trained personnel to interpret responses and produce the final assessment. Several efforts have attempted to reduce this burden by deriving shortened cognitive screening instruments. For example, Larner (2017) evaluated a short Montreal Cognitive Assessment (s-MoCA), which was derived from the original MoCA using item response theory and computerized adaptive testing. Such short-form instruments suggest that MoCA-style screening can be compressed while retaining useful clinical sensitivity. However, once derived, these instruments are still largely fixed-form: they do not dynamically select questions according to the subject's evolving response pattern, nor do they explicitly learn when to stop an ongoing interaction.

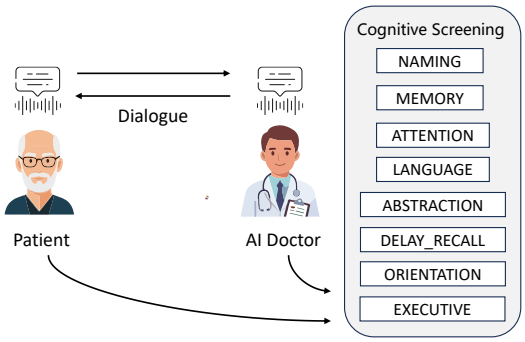


Figure 1: The AQS: The subject interacts with the adaptive question-answering system, which assesses the subject's cognitive 8-dimensional state based on their responses, intelligently selects the next question to ask based on the 8-dimensional state, and classifies and outputs the assessment results when the stop condition is met.

Language-based cognitive assessment has also attracted attention, especially methods that diagnose impairment from picture descriptions or speech transcripts (Slegers et al., 2018; Fraser et al., 2016; Luz et al., 2020). These methods capture richer linguistic cues than fixed questionnaires, but many are designed as one-shot classification tasks rather than interactive screening systems. Meanwhile, CAT methods improve efficiency through

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 19532–19548
October 24-29, 2026 ©2026 Association for Computational Linguistics

adaptive item selection, yet most CAT formulations assume discrete responses and are less suited to free-form natural language.

We propose AQS, an adaptive questioning system for cognitive impairment screening. As shown in Figure 1, AQS represents each subject with an 8-dimensional cognitive state derived from MoCA domains. At each turn, an LLM-based evaluator maps the subject's response into a discrete severity signal, the state updater revises the latent state, and the selector chooses the next question; a stopping module terminates the assessment once the diagnostic belief becomes sufficiently confident. Our main contributions include: (1) construction of a MoCA-grounded cognitive state model that represents each subject with an interpretable 8-dimensional latent profile for cognitive impairment screening; (2) development of a dynamic adaptive questioning framework that departs from MoCA's static fixed-item design, reducing the number of required questions while maintaining comparable diagnostic performance; and (3) distributional analysis showing that AQS yields a more balanced assessment process, with most subjects completed within a shorter question range and far fewer cases reaching the maximum budget.

2 Background

Adaptive Testing and Cognitive Modeling. Computerized Adaptive Testing (CAT) dynamically selects items based on previous responses, enabling more efficient assessment than fixed-item tests. It has been widely studied in educational testing, mental health assessment, and standardized evaluation (Wainer et al., 2000; Cheng, 2009; Gibbons et al., 2016; Varadarajan et al., 2024). A typical CAT system combines a Cognitive Diagnosis Model (CDM) for estimating latent traits with a Selection Algorithm (SA) for choosing the next item (Zhuang et al., 2022). Classical methods rely on Item Response Theory (IRT) and Fisher information (Embretson and Reise, 2000), while multidimensional models such as MIRT and G-DINA extend this formulation to multiple latent traits (Reckase, 2009; de la Torre, 2011). Recent neural CDMs further improve flexibility but often require large-scale interaction data (Wang et al., 2020, 2024), which is limited in cognitive impairment screening. We therefore adopt a parameterized and interpretable cognitive-state model. Most closely related to our clinical setting, Gibbons et al. (2024) proposed a computerized adaptive cognitive assessment based on multidimensional item response theory. Their framework calibrates a cognitive item bank with a MIRT model and demonstrates that adaptive administration can approximate full-bank cognitive scores with substantially fewer items. This work provides direct evidence that MIRT-based CAT is feasible for cognitive function assessment. In contrast, AQS focuses on an interactive language-based screening pipeline: free-form responses are converted into structured severity signals, latent cognitive states are updated online, and stopping decisions are learned under scarce real interaction trajectories.

Reinforcement Learning for Adaptive Assessment. RL provides a natural framework for adaptive testing by formulating question selection as a sequential decision-making problem. Prior work such as NCAT and BOBCAT shows that learned policies can outperform handcrafted item selection strategies (Zhuang et al., 2022; Ghosh and Lan, 2021), and recent surveys highlight RL as a promising direction for CAT systems (Liu et al., 2024). However, fully end-to-end RL is difficult in clinical screening, where real interaction trajectories and reliable patient-state annotations are scarce. In this work, we use RL as a policy optimization layer on top of a structured cognitive-state model rather than replacing the full assessment pipeline.

LLMs for Structured Response Evaluation. Large language models have shown strong ability in extracting psychological and clinically relevant signals from natural language, supporting tasks such as mental health assessment and cognitive diagnosis (Ji et al., 2022). However, reliable diagnosis from limited language input remains challenging (Harrigian et al., 2020). We therefore use LLMs not as standalone diagnostic models, but as evaluators that map free-form answers into structured severity signals for downstream cognitive-state estimation.

3 Methods

AQS consists of five key modules: a calibrated state updater, diagnostic classifier, patient simulator, adaptive selector, and stopping rule. The system maintains an 8-dimensional cognitive state; at each turn it asks one question, converts the response into a severity label, updates the state, predicts impairment risk, and either stops or continues.

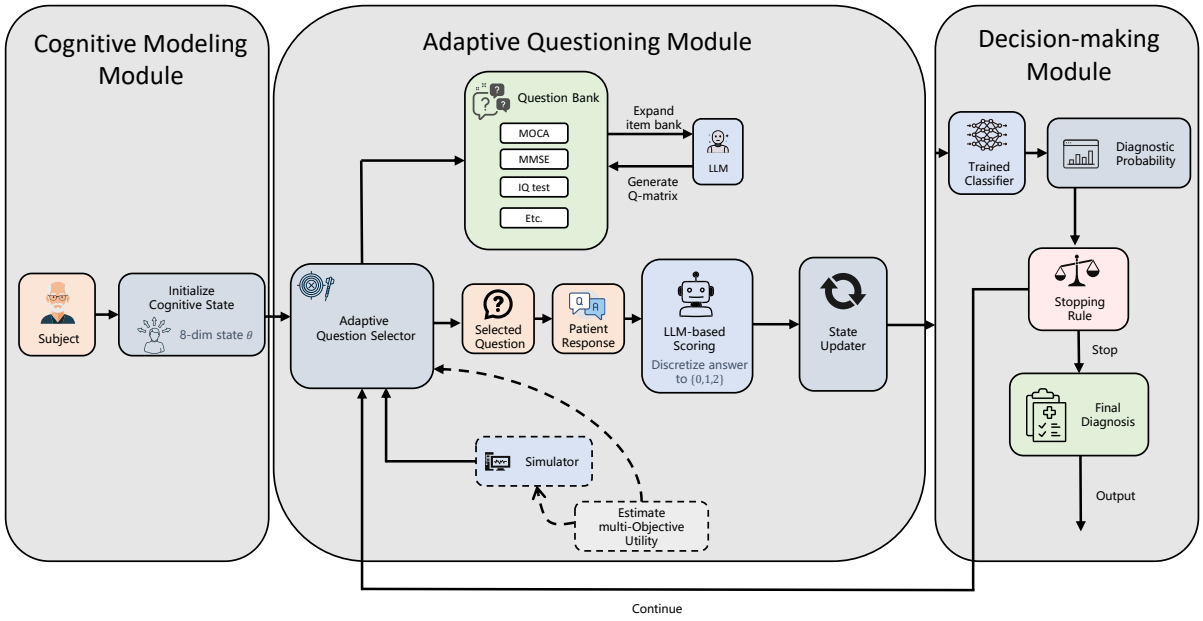


Figure 2: The system architecture of AQS.

Figure 2 summarizes this online loop.

3.1 Calibrated Cognitive State Updater

We represent the subject's cognitive condition at step t as an 8-dimensional latent state $\theta_t \in [0,1]^D$, where larger values indicate more severe impairment. After each interaction, the updater revises θ_t according to the selected question q_t, the response severity $v_t \in \{0,1,2\}$, and the confidence score $r_t \in [0,1]$.

Each question is associated with a cognitive-domain weight vector derived from a Q-matrix. To allow small data-driven corrections to the manually constructed Q-matrix, we define

$$\widetilde{Q} = \mathrm{Norm}\big(\mathrm{ReLU}(Q^{\text{base}} + \Delta Q)\big), \tag{1}$$

where $Q^{\text{base}} \in \mathbb{R}^{N\times D}$ is the base Q-matrix, ΔQ is a learnable correction term, and $\mathrm{Norm}(\cdot)$ denotes row-wise normalization. For the selected question, we use $w_{q_t} = \widetilde{Q}[q_t]$ as its calibrated domain-weight vector.

The discrete severity label is mapped to a continuous impairment target:

$$y_t = \begin{cases} 0, & v_t = 0, \\ \alpha_{\text{mid}}, & v_t = 1, \\ 1, & v_t = 2, \end{cases} \qquad \alpha_{\text{mid}} = \sigma(a_{\text{mid}}), \tag{2}$$

where the intermediate severity level is learned rather than fixed. We also calibrate the observation confidence by

$$c_t = \sigma(a_c r_t + b_c), \tag{3}$$

and learn a domain-wise update rate

$$\eta = \sigma(\eta_{\text{raw}}) \in (0,1)^D. \tag{4}$$

The latent state is updated using a gated residual rule:

$$\theta_{t+1} = \theta_t + \eta \odot c_t \odot w_{q_t} \odot (y_t - \theta_t), \tag{5}$$

followed by clipping to $[0,1]$. This update is bounded and interpretable: w_{q_t} determines which cognitive domains are affected, c_t controls the reliability of the observation, and η controls domain-wise update sensitivity.

To support online decision making, the updater is trained to reconstruct both final and prefix cognitive states from simulated trajectories. Given the ground-truth state θ^* and predicted prefix states $\hat{\theta}_t$, we optimize

$$\mathcal{L}_{\text{upd}} = \left\|\hat{\theta}_T - \theta^*\right\|_2^2 + \lambda_{\text{pre}} \frac{1}{T}\sum_{t=1}^{T} \left\|\hat{\theta}_t - \theta^*\right\|_2^2. \tag{6}$$

The final-state term encourages accurate diagnosis at termination, while the prefix term stabilizes intermediate state tracking during adaptive assessment.

3.2 Diagnostic Classifier

The diagnostic classifier maps the current latent state $\hat{\theta}_t$ to impairment probability and remains fixed during online assessment:

Formally, given the current latent state $\hat{\theta}_t$, the classifier outputs

$$p_t = f_{\text{cls}}(\hat{\theta}_t), \tag{7}$$

where $p_t \in [0, 1]$ denotes the estimated probability that the subject belongs to the impaired group. A binary prediction is then obtained by

$$\hat{y}_t = \mathbb{I}(p_t \geq \tau), \tag{8}$$

where τ is a learned decision threshold. Rather than fixing $\tau = 0.5$, we explicitly optimize it on grouped out-of-fold predictions to better balance screening sensitivity and specificity.

Model selection. We compare logistic regression, ℓ_1-regularized logistic regression, RBF-SVM, random forest, ExtraTrees, gradient boosting, and MLP, using `PTID` as the grouping variable. For each model, thresholds are selected by

$$\begin{aligned}\mathcal{J}_{\text{thr}} = 0.45 \cdot \text{BAcc} + 0.30 \cdot \text{F1} \\ + 0.15 \cdot \text{Acc} + 0.10 \cdot \text{AUC}.\end{aligned} \tag{9}$$

The best trainable model is then refit on the full training set. As shown in Table 1, ℓ_1-regularized logistic regression achieves the best overall trainable-model performance and is used as the diagnostic backbone.

Model	Acc	BAcc	F1	Sens	AUC
LogReg-ℓ_1	**0.736**	**0.736**	0.735	0.741	**0.810**
LogReg	0.735	0.736	0.735	0.741	0.810
SVM-RBF	0.732	0.732	0.724	0.713	0.778
GBDT	0.722	0.723	0.727	0.748	0.798
MOCA rule	0.716	0.717	**0.739**	**0.811**	0.806
ExtraTrees	0.713	0.714	0.731	0.789	0.789
Random Forest	0.713	0.713	0.726	0.770	0.784
MLP	0.694	0.694	0.700	0.723	0.749

Table 1: Out-of-fold performance of diagnostic classifiers on the training split.

3.3 Patient Simulator

Because large-scale free-form trajectories from cognitively impaired patients are unavailable, we use a parameterized simulator to generate question-level observations for policy learning. Given a latent cognitive state $\theta \in [0, 1]^D$ and a selected question q, the simulator returns a severity label $s \in \{0, 1, 2\}$ and a confidence score $r \in [0, 1]$, where larger values of θ indicate more severe impairment.

We instantiate the simulator as a lightweight multidimensional graded-response model (Samejima, 1969; Reckase, 2009). Let $w_q \in \mathbb{R}^D$ be the normalized Q-matrix row of question q. The item-specific impairment score is

$$z_q = \langle w_q, \theta \rangle. \tag{10}$$

Each item has a discrimination parameter a_q and two ordered thresholds $(b_{q,1}, b_{q,2})$. The cumulative probabilities are

$$P(s \geq k \mid q, \theta) = \sigma\big(a_q(z_q - b_{q,k})\big), \quad k \in \{1, 2\}, \tag{11}$$

which define

$$\begin{aligned}P(s = 0) &= 1 - P(s \geq 1), \\ P(s = 1) &= P(s \geq 1) - P(s \geq 2), \\ P(s = 2) &= P(s \geq 2).\end{aligned} \tag{12}$$

We additionally include small guess and slip probabilities to avoid deterministic responses. The final severity is sampled from the adjusted categorical distribution, and the confidence score is computed from distribution sharpness.

Simulator Faithfulness. We validate the simulator against empirical MoCA-derived severity patterns. For each cognitive domain, we compare real and simulated three-level severity distributions using total variation distance:

$$\text{TV}(P_{\text{real}}, P_{\text{sim}}) = \frac{1}{2} \sum_{s=0}^{2} |P_{\text{real}}(s) - P_{\text{sim}}(s)|. \tag{13}$$

As shown in Figure 3, the simulator closely follows the empirical domain-level distributions, with an average TV distance of 0.085. The largest deviations occur in Memory/Learning and Delayed Recall, where severe responses are more concentrated in the real data.

We further examine whether the simulator preserves subject-level domain ordering. Table 2 shows high Pearson correlations across domains, indicating that subjects with stronger empirical impairment generally receive higher simulated severity. Discretized correlations remain strong for most domains, especially Memory/Learning, Delayed Recall, and Abstraction, supporting the simulator's use for policy learning.

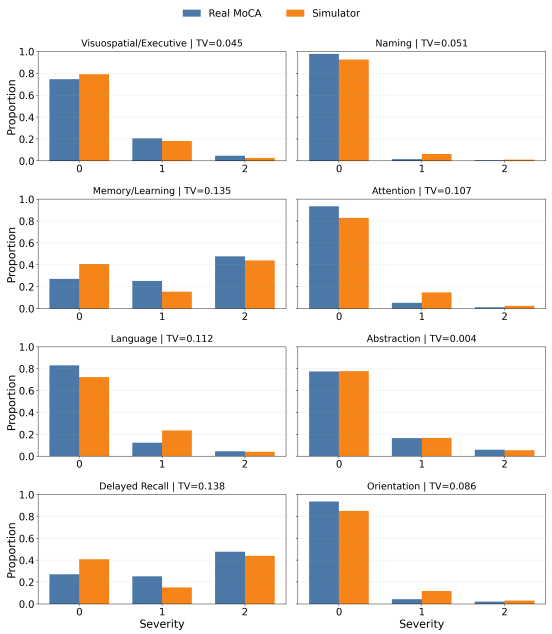


Figure 3: Domain-level severity distributions of real MoCA records and simulator-generated responses. TV denotes total variation distance.

Domain	Pear.	Spear.	Pear.-D	Spear.-D
Vis./Exec.	0.920	0.903	0.812	0.680
Naming	0.946	0.634	0.778	0.679
Memory	0.980	0.959	0.944	0.954
Attention	0.949	0.795	0.868	0.712
Language	0.943	0.903	0.833	0.759
Abstraction	0.979	0.729	0.999	1.000
Delayed Rec.	0.980	0.960	0.944	0.953
Orientation	0.936	0.682	0.870	0.727

Table 2: Correlation between real MoCA-derived impairment and simulator-generated severity. "-D" denotes correlations after discretization.

3.4 Adaptive Selection Algorithm

At each step, AQS combines a multi-objective question selector with a fixed stopping rule. For each remaining item, the selector simulates possible outcomes, updates the latent state, and scores the expected diagnostic utility:

$$\begin{aligned}\mathrm{Score}(q) = \lambda_{\mathrm{info}} U_{\mathrm{info}}(q) - \lambda_{\theta} U_{\theta}(q) \\ - \lambda_{\mathrm{risk}} U_{\mathrm{risk}}(q) - \lambda_{\mathrm{cost}} U_{\mathrm{cost}}(q) \\ + \lambda_{\mathrm{drift}} U_{\mathrm{drift}}(q).\end{aligned} \tag{14}$$

Here U_{info}, U_{θ}, U_{risk}, U_{cost}, and U_{drift} capture informativeness, state consistency, diagnostic risk, question cost, and useful state movement, respectively. The system asks

$$q_t = \arg\max_{q \in \mathcal{Q}_t} \mathrm{Score}(q), \tag{15}$$

After each update, the fixed stop rule terminates when $t \geq T_{\max}$ or when $t \geq T_{\min}$ and p_t crosses either the high-risk or low-risk threshold. This handcrafted policy serves as the main structured baseline.

3.5 Reinforcement Learning Formulation

Since no single selection strategy dominates across budgets and metrics: adaptive selection is strong at moderate budgets, and sequential selection favors sensitivity. This motivates an RL formulation that optimizes long-term diagnostic utility rather than a single handcrafted criterion.

MDP formulation. We formulate adaptive screening as an MDP. The RL state is denoted by s_t:

$$s_t = \left[\hat{\theta}_t \,;\, p_t \,;\, H(p_t) \,;\, m_t \,;\, \rho_t\right], \tag{16}$$

where $\hat{\theta}_t$ is the latent state, p_t the impairment probability, $H(p_t)$ its entropy, m_t the asked-question mask, and $\rho_t = t/T_{\max}$ the normalized step. Actions either ask a valid remaining question or stop; invalid actions are masked. Question actions trigger simulation, state updating, and reclassification, while stop actions terminate the episode.

Reward design. The reward balances diagnostic quality and efficiency: each question incurs cost, correct stopping receives positive reward, and false negatives are penalized more than false positives:

$$r_t = \begin{cases} -c_q, & \text{ask}, \\ R_{\mathrm{cor}} + R_{\mathrm{early}}, & \text{correct stop}, \\ -R_{\mathrm{FP}}, & \text{false positive}, \\ -R_{\mathrm{FN}}, & \text{false negative}. \end{cases} \tag{17}$$

where c_q is the question cost, R_{early} is an optional bonus for timely stopping, and $R_{\mathrm{FN}} > R_{\mathrm{FP}}$ reflects the higher cost of missed impairment.

Training algorithm. We train policies with Double DQN (van Hasselt et al., 2016) using replay-buffer sampling, target networks, action masking, and optional behavior-cloning warm starts. The temporal-difference loss is

$$\mathcal{L}_{\mathrm{DQN}} = \mathbb{E}_{\tau_t \sim \mathcal{B}}\left[\left(y_t - Q_{\phi}(s_t, a_t)\right)^2\right], \tag{18}$$

where the Double DQN target is

$$y_t = r_t + \gamma(1 - d_t)Q_{\phi^-}\Big(s_{t+1}, \arg\max_{a' \in \mathcal{A}_{t+1}} Q_\phi(s_{t+1}, a')\Big). \tag{19}$$

Here, $\mathcal{B}$ denotes the replay buffer, ϕ^- denotes the target network parameters, and $\mathcal{A}_{t+1}$ is the set of valid next actions after masking invalid choices. To stabilize optimization, we additionally use teacher-generated warm-start trajectories and optional behavior cloning pretraining before online DQN updates.

Three RL settings. We consider three RL settings that differ in which component is learned. The system model of the three-component reinforcement learning is shown in Figure 4.

Question-only. In this setting, RL is used only for question selection, while stopping is governed by the fixed stopping rule described above. The action space contains the remaining candidate questions, and the policy learns to maximize long-term diagnostic utility under a fixed stop mechanism.

Stop-only. In this setting, question selection is fixed to the adaptive handcrafted selector, and RL only decides whether to CONTINUE or STOP. This isolates the stopping problem from question selection and allows the agent to optimize the trade-off between assessment length and diagnostic reliability.

Joint. In the joint setting, RL is used to learn both question selection and stopping in a unified framework. This is the most expressive formulation, but also the most difficult to optimize because the question policy and the stopping policy can influence each other through the evolving state distribution.

4 Dataset

4.1 Patient Cohort and Preprocessing

The patient records used in this study were derived from the **Alzheimer's Disease Neuroimaging Initiative (ADNI)** dataset(Weiner et al., 2010), containing cognitive assessment data for normal cognition (NC), mild cognitive impairment (MCI), and Alzheimer's disease (AD) subjects. We extracted subject-level MoCA assessment records together with diagnostic labels. For the sake of convenience in statistics, we have combined MCI and AD into one category, and both are regarded as having a risk of cognitive impairment.

Following the MoCA protocol, each subject was represented by an 8-dimensional latent cognitive state:

$$\theta = \big[\theta_{\text{vis}}, \theta_{\text{name}}, \theta_{\text{mem}}, \theta_{\text{att}}, \theta_{\text{lang}}, \theta_{\text{abs}}, \theta_{\text{delay}}, \theta_{\text{ori}}\big]. \tag{20}$$

4.2 Question Bank

We constructed an expanded cognitive question bank based on standardized clinical assessment scales, including MoCA, MMSE, and selected Wechsler-style cognitive tasks(Nasreddine et al., 2005; Folstein et al., 1975). To improve conversational diversity, a medical large language model (LLM) was further used to generate semantically equivalent question variants under cognitive-dimension constraints.

4.3 Dataset Splits and Usage

To ensure strict held-out evaluation, all model development is performed on $\mathcal{D}_A$, while the unseen $\mathcal{D}_B$ is used only for final system evaluation; the detailed usage of each split is summarized in Table 3.

Subset	# Patients	Usage
$\mathcal{D}_A$	2,385	Classifier and policy development
$\mathcal{D}_A^{\text{train}}$	2,086	RL rollout generation and policy learning
$\mathcal{D}_A^{\text{valid}}$	299	Checkpoint selection and tuning
$\mathcal{D}_B$	597	Final held-out evaluation

Table 3: Dataset splits and usage. $\mathcal{D}_A$ is used for all model development, while $\mathcal{D}_B$ is kept unseen until final evaluation.

4.4 RL Trajectory Construction

Since existing clinical datasets do not contain adaptive conversational trajectories, we constructed an RL interaction dataset using the proposed patient simulator and adaptive assessment environment.

For each subject, the simulator initialized the latent cognitive state according to the corresponding cognitive profile. The agent then sequentially interacted with the environment by selecting questions, receiving simulated responses, updating the latent state, and obtaining classifier predictions.

To increase trajectory diversity, interaction episodes were generated using multiple behavioral policies, including random selection, sequential selection, and the handcrafted adaptive selector. Each trajectory is represented as

$$\tau = (s_0, a_0, R_0, \ldots, s_T), \tag{21}$$

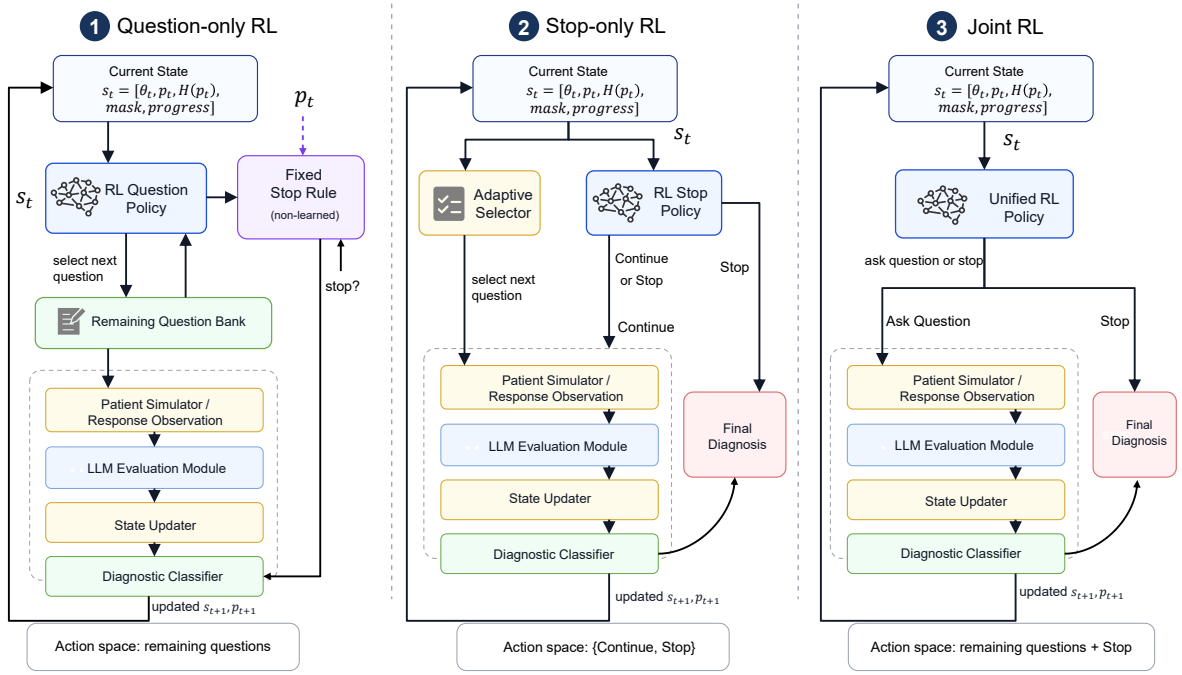


Figure 4: The system process of three reinforcement learning strategies.

where s_t, a_t, and R_t denote the RL state, action, and reward at step t, respectively.

5 Results and Discussion

5.1 Overall Performance Comparison

We first compare different combinations of question selection and stopping strategies, including Sequential, Random, FisherInfo Selector and Adaptive Selector[1], RL Question Selector + Fixed Stop, RL Question Selector + RL Stop, and Adaptive Selector + RL Stop. Figure 5 reports the overall diagnostic performance and assessment cost.

Several observations follow from Table 4. First, purely heuristic baselines such as Sequential and Random require substantially more questions to achieve competitive performance. Second, RL-based stopping greatly reduces the assessment length. Most importantly, Adaptive Selector + RL Stop delivers the best overall system performance. It achieves the highest BAcc and AUC while reducing AvgQ from 13.94 to 9.78 relative to Adaptive Selector.

The relatively weak performance of the joint RL policy should not be interpreted as evidence that joint optimization is inherently inferior. In principle, a unified policy that jointly learns question selection and stopping could discover globally optimal assessment strategies. However, under our current simulator, reward design, and limited offline trajectories, the two objectives create strong optimization tension: the question policy tends to acquire more evidence for better discrimination, whereas the stop policy is rewarded for shortening the assessment. This conflict makes naive joint DQN training unstable. We therefore treat the joint result as a useful negative result under the current formulation, and leave more stable joint optimization, such as hierarchical RL, constrained policy optimization, or multi-objective RL, as future work.

5.2 Question-Count Distribution Analysis

Average question count alone cannot determine whether a policy is genuinely efficient, since a low AvgQ may result from either balanced assessment or pathological early stopping. Therefore, Table 5 further reports the distribution of question counts, Figure 6 can more clearly show the differences in the distribution of the number of questions stopped among different strategies. Adaptive Selector shows a bimodal pattern, with many subjects stopping early but many others reaching the longest range, indicating that the fixed stop rule remains conservative for difficult cases. In contrast, Adaptive Selector + RL Stop concentrates most subjects in the 5–10 question range and substantially reduces the 25–30 tail.

[1]The four basic strategies mentioned here and later are all based on Fixed stop.

Policy	Acc	BAcc	F1	Sens	Spec	AUC	AvgQ
Random	0.686 ± 0.004	0.689 ± 0.004	0.704 ± 0.004	0.782 ± 0.011	0.597 ± 0.014	0.756 ± 0.010	23.109 ± 0.377
Sequential	0.681 ± 0.005	0.687 ± 0.005	0.720 ± 0.006	0.857 ± 0.014	0.518 ± 0.008	0.763 ± 0.015	16.475 ± 0.256
FisherInfo Selector	0.682 ± 0.009	0.689 ± 0.009	**0.722 ± 0.007**	**0.862 ± 0.008**	0.517 ± 0.010	0.762 ± 0.007	14.435 ± 0.539
Adaptive Selector	0.708 ± 0.009	0.709 ± 0.009	0.709 ± 0.010	0.741 ± 0.016	0.678 ± 0.011	0.751 ± 0.016	13.941 ± 0.273
Adaptive Selector+RL Stop	**0.715 ± 0.014**	**0.716 ± 0.014**	0.715 ± 0.014	0.748 ± 0.016	**0.684 ± 0.015**	**0.771 ± 0.012**	9.778 ± 0.227
RL Selector+Fixed Stop	0.689 ± 0.012	0.695 ± 0.012	0.720 ± 0.010	0.836 ± 0.011	0.554 ± 0.019	0.750 ± 0.011	21.196 ± 0.171
RL Selector+RL Stop	0.669 ± 0.015	0.674 ± 0.015	0.698 ± 0.014	0.797 ± 0.020	0.552 ± 0.022	0.756 ± 0.016	**5.854 ± 0.077**

Table 4: Multi-seed comparison of policy and stopping-rule combinations on the full test set. Results are reported as mean ± standard deviation across random seeds. The best value in each column is shown in bold. AvgQ denotes the average number of questions used during assessment.

Method	0–5	5–10	10–15	15–20	20–25	25–30
Random	1.8	9.4	12.3	9.4	8.5	58.6
Sequential	6.4	29.0	14.3	14.3	3.1	33.0
FisherInfo Selector	33.4	20.0	4.7	14.3	3.1	24.5
Adaptive Selector	40.7	13.3	7.9	5.7	3.3	29.1
Adaptive Selector + RL Stop	19.8	**45.3**	13.2	12.0	3.4	**6.3**
RL Selector + Fixed Stop	0.0	23.8	14.8	10.1	5.7	45.5
RL Selector + RL Stop	14.1	1.6	1.0	23.2	26.6	33.4

Table 5: Binned distribution (%) of question counts under different policies. Adaptive Selector + RL Stop concentrates the largest proportion of subjects in the 5–10 question range while sharply reducing the proportion of long assessments in the 25–30 range.

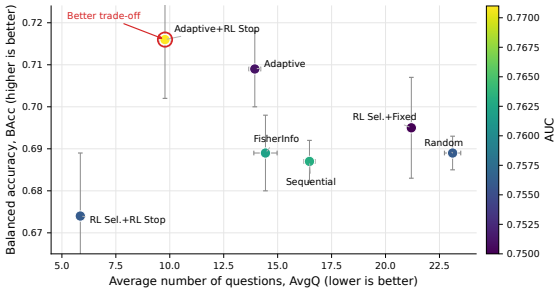


Figure 5: Performance-efficiency balance of different policies.

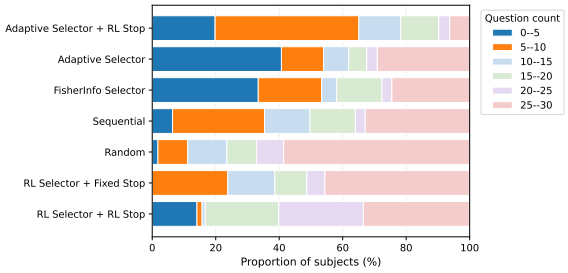


Figure 6: Question-count distribution under different policies.

6 Conclusion

In this paper, we proposed an adaptive cognitive screening framework that integrates parameterized cognitive state modeling, LLM-based response evaluation, and reinforcement learning-based policy optimization. The framework maintains an explicit 8-dimensional latent cognitive state, updates this state online from question-level observations, and dynamically determines both what to ask and when to stop. This design enables more efficient cognitive assessment than fixed-length screening while preserving interpretability under data-scarce medical settings.

Our experiments demonstrate a strong performance-efficiency trade-off. In particular, combining the handcrafted adaptive selector with an RL-based stopping policy achieves the best overall results, improving diagnostic performance while substantially reducing the average number of questions. Further comparisons among question-only, stop-only, and joint RL settings suggest that question selection and stopping are better treated as two decoupled decision sub-problems: both can benefit from reinforcement learning when optimized separately, whereas naive joint optimization leads to unstable training and inferior performance.

Overall, our findings suggest that structured cognitive priors remain crucial for robust question selection, while reinforcement learning is especially effective for learning dynamic stopping behaviors. More broadly, this work highlights the value of combining interpretable latent-state modeling with learned sequential decision policies for adaptive medical assessment.

Limitations

Our work has several limitations. First, the reinforcement learning component is trained primarily on simulator-generated interaction trajectories rather than large-scale real conversational screening data. Although the simulator is structurally grounded in the latent cognitive state and Q-matrix, it remains an approximation of real patient behavior. As a result, the learned policies may not fully capture the diversity and complexity of natural responses in clinical settings.

The response evaluation module simplifies free-form language into a discrete severity label in $\{0, 1, 2\}$. This abstraction makes the adaptive pipeline tractable and compatible with the latent-state updater, but it inevitably compresses rich linguistic information. In particular, subtle pragmatic cues, hesitation patterns, and partial semantic correctness may not be fully preserved under this discretization.

While we systematically compared *question-only*, *stop-only*, and *joint* reinforcement learning formulations, our current conclusions are tied to the present simulator, reward design, and training setup. In particular, the observed weakness of joint RL should be interpreted as evidence under the current formulation rather than a universal claim that joint optimization is always inferior for adaptive screening.

Finally, our experiments focus on one cognitive screening setting derived from ADNI-based assessment data. The generalizability of the proposed framework to other patient populations, languages, assessment protocols, or clinical environments remains to be validated.

Ethical Considerations

This work studies adaptive cognitive screening under data-scarce medical settings. Although the proposed framework is motivated by clinical cognitive assessment, it is intended as an offline research prototype rather than a deployable diagnostic system. The model outputs should not be interpreted as clinical diagnoses, and any real-world use would require prospective validation, clinical supervision, and approval by relevant medical and ethical review boards.

The patient records used in this study are derived from ADNI-based cognitive assessment data. We use only structured assessment records and diagnostic labels for model development and evaluation, and do not attempt to identify individual participants. All experiments are conducted at the aggregate level. At the same time, in the dataset, only the subject's ID is provided and no specific identity information is included. If applied to real conversational screening in the future, stronger privacy protections would be necessary, since free-form responses may contain sensitive personal or health-related information.

References

Ying Cheng. 2009. When cognitive diagnosis meets computerized adaptive testing: CD-CAT. *Psychometrika*, 74(4):619–632.

Jimmy de la Torre. 2011. The generalized DINA model framework. *Psychometrika*, 76(2):179–199.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Susan E. Embretson and Steven P. Reise. 2000. *Item Response Theory for Psychologists*. Lawrence Erlbaum Associates, Mahwah, NJ.

Marshal F. Folstein, Susan E. Folstein, and Paul R. McHugh. 1975. "Mini-mental state": A practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research*, 12(3):189–198.

Kathleen C. Fraser, Jed A. Meltzer, and Frank Rudzicz. 2016. Linguistic features identify alzheimer's disease in narrative speech. *Journal of Alzheimer's Disease*, 49(2):407–422.

Aritra Ghosh and Andrew Lan. 2021. BOBCAT: Bilevel optimization-based computerized adaptive testing. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 2410–2417.

Robert D. Gibbons, Diane S. Lauderdale, Robert S. Wilson, David A. Bennett, Tesnim Arar, and David A. Gallo. 2024. Adaptive measurement of cognitive function based on multidimensional item response theory. *Alzheimer's & Dementia: Translational Research & Clinical Interventions*, 10(4):e70018.

Robert D. Gibbons, David J. Weiss, Ellen Frank, and David Kupfer. 2016. Computerized adaptive diagnosis and testing of mental health disorders. *Annual Review of Clinical Psychology*, 12:83–104.

Keith Harrigian, Carlos Aguirre, and Mark Dredze. 2020. Do models of mental health based on social media data generalize? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3774–3788, Online. Association for Computational Linguistics.

Shaoxiong Ji, Tianlin Zhang, Luna Ansari, Jie Fu, Prayag Tiwari, and Erik Cambria. 2022. MentalBERT: Publicly available pretrained language models for mental healthcare. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7184–7190, Marseille, France. European Language Resources Association.

Andrew J. Larner. 2017. Short Montreal Cognitive Assessment: Validation and reproducibility. *Journal of Geriatric Psychiatry and Neurology*, 30(2):104–108.

Qi Liu, Yan Zhuang, Haoyang Bi, Zhenya Huang, Weizhe Huang, Jiatong Li, Junhao Yu, Zirui Liu, Zirui Hu, Yuting Hong, Zachary A. Pardos, Haiping Ma, Mengxiao Zhu, Shijin Wang, and Enhong Chen. 2024. Survey of computerized adaptive testing: A machine learning perspective. *arXiv preprint arXiv:2404.00712*.

Saturnino Luz, Fasih Haider, Sofia de la Fuente, Davida Fromm, and Brian MacWhinney. 2020. Alzheimer's dementia recognition through spontaneous speech: The ADReSS challenge. In *Proceedings of Interspeech 2020*, pages 2172–2176.

Ziad S. Nasreddine, Natalie A. Phillips, Valérie Bédirian, Simon Charbonneau, Victor Whitehead, Isabelle Collin, Jeffrey L. Cummings, and Howard Chertkow. 2005. The Montreal Cognitive Assessment, MoCA: A brief screening tool for mild cognitive impairment. *Journal of the American Geriatrics Society*, 53(4):695–699.

Mark D. Reckase. 2009. *Multidimensional Item Response Theory*. Springer, New York.

Fumiko Samejima. 1969. Estimation of latent ability using a response pattern of graded scores. *Psychometrika*, 34(S1):1–97.

Antoine Slegers, Renée-Pier Filiou, Maxime Montembeault, and Simona Maria Brambati. 2018. Connected speech features from picture description in alzheimer's disease: A systematic review. *Journal of Alzheimer's Disease*, 65(2):519–542.

Baichuan-M3 Team, Chengfeng Dou, Fan Yang, Fei Li, Jiyuan Jia, Qiang Ju, Shuai Wang, Tianpeng Li, Xiangrong Zeng, Yijie Zhou, Hongda Zhang, Jinyang Tai, Linzhuang Sun, Peidong Guo, Yichuan Mo, Xiaochuan Wang, Hengfu Cui, and Zhishou Zhang. 2026. Baichuan-m3: Modeling clinical inquiry for reliable medical decision-making. *arXiv preprint*, abs/2602.06570.

Hado van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double Q-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, pages 2094–2100.

Vasudha Varadarajan, Sverker Sikström, Oscar N. E. Kjell, and H. Andrew Schwartz. 2024. ALBA: Adaptive language-based assessments for mental health. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2466–2478, Mexico City, Mexico. Association for Computational Linguistics.

Howard Wainer, Neil J. Dorans, Ronald Flaugher, Bert F. Green, Robert J. Mislevy, Lynne Steinberg, and David Thissen, editors. 2000. *Computerized Adaptive Testing: A Primer*, 2 edition. Lawrence Erlbaum Associates, Mahwah, NJ.

Fei Wang, Weibo Gao, Qi Liu, Jiatong Li, Guanhao Zhao, Zheng Zhang, Zhenya Huang, Mengxiao Zhu, Shijin Wang, Wei Tong, and Enhong Chen. 2024. A survey of models for cognitive diagnosis: New developments and future directions. *arXiv preprint arXiv:2407.05458*.

Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. 2020. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6153–6161.

Michael W. Weiner, Paul S. Aisen, Clifford R. Jack, William J. Jagust, John Q. Trojanowski, Leslie Shaw, Andrew J. Saykin, John C. Morris, Nigel Cairns, Laurel A. Beckett, Arthur Toga, Robert Green, Sarah Walter, Holly Soares, Peter Snyder, Eric Siemers, William Potter, Philip E. Cole, and Michael Schmidt. 2010. The alzheimer's disease neuroimaging initiative: Progress report and future plans. *Alzheimer's & Dementia*, 6(3):202–211.e7.

Yan Zhuang, Qi Liu, Zhenya Huang, Zhi Li, Shuanghong Shen, and Haiping Ma. 2022. Fully adaptive framework: Neural computerized adaptive testing for online education. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4734–4742.

A LLM-based Response Evaluation

The response evaluation module maps each free-form response into a discrete severity label $s \in \{0, 1, 2\}$ and a confidence score $r \in [0, 1]$, where 0, 1, and 2 denote normal, mildly suspicious, and clearly abnormal performance, respectively. The full prompt template is provided at the end of this appendix. During evaluation, we use deterministic decoding with temperature 0.0. Model outputs are parsed by extracting JSON-like spans containing the `severity` field. If parsing fails, we apply a fallback parser; if this also fails, we assign the default uncertain label `severity` $= 1$ and `confidence` $= 0.5$.

To validate this module, we construct a rubric-controlled response evaluation set from the expanded question bank. DeepSeek(DeepSeek-AI, 2024) is used to generate free-form answers for

Evaluator	Acc	Macro-F1	κ
Baichuan w/ criterion + few-shot	0.773	0.756	0.559
Baichuan w/o criterion	0.807	0.798	0.611
Baichuan w/ criterion	**0.866**	**0.864**	**0.698**
Evaluator	QWK	MAE	
Baichuan w/ criterion + few-shot	0.819	0.259	
Baichuan w/o criterion	0.855	0.219	
Baichuan w/ criterion	**0.891**	**0.151**	

Table 6: Performance of Baichuan-M3-Plus response evaluation on rubric-controlled free-form answers generated by DeepSeek. QWK denotes quadratic weighted kappa.

each question under three intended severity levels, yielding 960 responses in total. The intended severity is treated as a silver label, so this experiment evaluates rubric-following consistency rather than clinical validity.

As shown in Table 6, providing item-specific scoring criteria substantially improves response evaluation. The criterion-aware evaluator achieves 0.866 accuracy, 0.864 macro-F1, and 0.891 QWK, outperforming the criterion-free variant by 5.8 accuracy points and 6.5 macro-F1 points. We therefore use the zero-shot criterion-aware setting as the default response evaluator in AQS.

B Question Bank and Q-matrix Construction

B.1 Question Bank Summary

The expanded question bank contains 80 items grouped into eight cognitive domains. Each item includes a question identifier, natural-language instruction, item type, difficulty value, and scoring criterion. Table 7 summarizes the domain-level composition. The complete item bank and the compact Q-matrix summary are collected at the end of this appendix in Tables 11 and 12, respectively.

B.2 Q-matrix Annotation

Each question was annotated with its corresponding cognitive dimensions to construct the Q-matrix:

$$Q \in [0,1]^{N \times D}, \tag{22}$$

where $Q_{ij} = 1$ indicates that question i involves cognitive dimension j.

Domain	#Items	Min Diff.	Max Diff.
Vis./Exec.	15	0.25	0.90
Naming	8	0.30	0.65
Memory	2	0.50	0.65
Attention	17	0.35	0.90
Language	16	0.20	0.85
Abstraction	14	0.35	0.95
Delayed Recall	1	0.60	0.60
Orientation	7	0.40	0.60

Table 7: Summary of the expanded cognitive question bank.

The annotation script loads question records, builds question-specific prompts, asks Baichuan-M3-Plus(Team et al., 2026)annotators to output JSON-formatted weights, validates and normalizes the weights, and aggregates valid annotations using a confidence-weighted average. The full annotation prompt is provided at the end of the appendix in Section E.3.

The generated Q-matrix annotations were manually reviewed by biomedical engineering researchers according to cognitive-domain definitions and item-level scoring criteria.

C Adaptive Selector Tuning

The handcrafted adaptive selector contains several weights controlling the relative importance of information gain, latent-state consistency, diagnostic risk, question cost, and positive diagnostic drift. We tune these parameters by grid search on the evaluation cohort. Specifically, we define a small set of candidate selector profiles, including conservative and balanced configurations, and combine each profile with a classifier decision threshold selected from a predefined threshold grid. When an RL stopping controller is available, each selector configuration is evaluated under both fixed stopping and RL-based stopping.

Each candidate selector is evaluated by running full adaptive assessment episodes. We rank candidate configurations by a validation selection score and use balanced accuracy, AUC, F1, and specificity as tie-breaking criteria. This procedure selects the adaptive selector parameters used in the subsequent policy comparison experiments. The best overall configuration in the current version is shown in Table 8.

Parameter	Value
Classifier threshold	0.58
λ_{info}	0.95
λ_{θ}	1.00
λ_{risk}	0.25
λ_{cost}	0.10
λ_{drift}	0.28
False-positive cost C_{FP}	1.90
False-negative cost C_{FN}	1.35
Low-risk stop threshold	0.25
High-risk stop threshold	0.65

Table 8: Best adaptive selector configuration selected by parameter search.

D Hyperparameters of RL

We train all RL policies with Double DQN using a target network, replay buffer, and hard action masking. The Q-network is a two-layer MLP with 256 hidden units per layer. Invalid actions, including repeated questions and premature stopping, are masked during both action selection and target computation.

Checkpoints are selected on the validation set using a utility that balances F1 and assessment efficiency:

$$\mathcal{U}_{\text{valid}} = \text{F1} - \lambda_q \frac{\text{AvgQ}}{T_{\max}} - \lambda_{\text{hit}} \text{MaxHit}, \quad (23)$$

The detailed hyperparameters are listed in Tables 10 and 9.

Reward / selection parameter	Value
Base question cost	0.10
Question-only question cost	0.00
Stop-only question cost	0.03
False-positive cost	0.80
False-negative cost	1.50
Correct early-stop bonus	0.80
Stop-only early-stop bonus	0.05
Forced-stop penalty	0.80
High-risk stop threshold	0.65
Low-risk stop threshold	0.30
Reward clipping	2.0
TD target clipping	20.0
Utility penalty λ_q	0.12
Utility penalty λ_{hit}	0.20
Utility tie tolerance	1×10^{-4}

Table 9: Reward and checkpoint-selection parameters used in RL training.

Hyperparameter	Value
RL algorithm	Double DQN
Q-network	MLP, 256–256 hidden units
Optimizer	Adam
Learning rate	1×10^{-4}
Discount factor γ	0.98
Batch size	512
Replay buffer size	200,000
Warm-up transitions	5,000
Exploration schedule	$\epsilon : 1.0 \rightarrow 0.05$
Epsilon decay steps	50,000
Target update interval	1,000 steps
Gradient clipping	5.0
Training episodes	100,000
Maximum questions $T_{\max}$	30
Minimum stop step $T_{\min}$	4
Validation interval	5000 episodes
Logging interval	500 episodes
BC pretraining epochs	3

Table 10: Main hyperparameters for Double-DQN policy learning.

E Full Tables and Prompt Templates

E.1 Complete Question Bank

Domain	Question ID	Type	Diff.	Instruction
Vis./Exec.	vis_analog_1	logic_analogy	0.25	A hat is worn on the head. Where are shoes worn?
Vis./Exec.	vis_analog_2	logic_analogy	0.30	Birds fly in the sky. Where do fish swim?
Vis./Exec.	vis_logic_seq_1	logic_sequence	0.35	Listen to this number pattern: 2, 4, 6, 8. What should the next number be?
Vis./Exec.	vis_logic_seq_2	logic_sequence	0.35	Listen to this pattern: 10, 9, 8, 7, 6. What comes next?
Vis./Exec.	vis_cube	cube_description	0.40	Look at this cube. Please describe roughly how many faces and edges it has, and what kind of three-dimensional shape it is.
Vis./Exec.	vis_clock_8_20	clock_mental_8_20	0.45	Imagine a clock showing 8:20. Please describe where the minute hand and hour hand would approximately point.
Vis./Exec.	vis_logic_seq_3	logic_sequence	0.45	Listen to this pattern: 1, 3, 5, 7. What comes next?
Vis./Exec.	vis_trails	trails_auditory	0.50	Please say the numbers and letters in this alternating order: 1-A, 2-B, 3-C.
Vis./Exec.	vis_clock_11_10	clock_mental_11_10	0.60	Imagine a round clock showing 11:10. Please describe where the hour hand and minute hand would approximately point.
Vis./Exec.	vis_analog_3	logic_analogy	0.60	A tree is to leaves as a bird is to what? Hint: what covers its body.
Vis./Exec.	vis_logic_seq_4	logic_sequence	0.65	Listen to this pattern: 2, 4, 8, 16. What comes next?
Vis./Exec.	vis_analog_4	logic_analogy	0.70	A grandfather is to a grandson as an uncle is to what?
Vis./Exec.	vis_map_route	map_route_mental	0.75	Imagine a simple community map: the entrance is at the bottom, a garden is in the middle, a supermarket is at the upper left, and a clinic is at the upper right. If you enter through the gate and go first to the supermarket and then to the clinic, please describe the route.
Vis./Exec.	vis_logic_seq_5	logic_sequence	0.85	Listen to this pattern: 1, 2, 4, 7, 11. What comes next?
Vis./Exec.	vis_logic_seq_fib	logic_sequence	0.90	Listen to this pattern: 1, 1, 2, 3, 5, 8. What comes next?
Naming	naming_def_1	naming_definition	0.30	Name an animal with a long trunk and large ears that is the largest land animal.
Naming	naming_body_parts	naming_body_parts	0.35	Describe where the nose is on the face, and where the ears and mouth are located on the head.
Naming	naming_def_3	naming_definition	0.40	Name a tool with two blades connected by a central pivot, used to cut paper or cloth.
Naming	naming_animals	naming_animals	0.50	Name the animals shown in the picture.
Naming	naming_def_2	naming_definition	0.50	Name an object used to measure temperature, often containing mercury or alcohol, or showing digital numbers.
Naming	naming_objects	naming_objects	0.55	Name the objects shown in the picture.
Naming	naming_def_5	naming_definition	0.60	Name the natural phenomenon: a colorful arc that appears in the sky after rain.
Naming	naming_def_4	naming_definition	0.65	Name the occupation: this person works in a court and makes legal judgments, often using a gavel.
Memory	memory_learning_5words	memory_learning	0.50	I will read five words. Please listen carefully and try to remember them: face, velvet, church, daisy, red. After I finish, please say as many words as you can remember.
Memory	memory_short_story	memory_story_immediate	0.65	I will tell you a short story. After listening, please retell it in your own words: Mr. Wang went to the market in the morning to buy apples and eggs, and on his way back he picked up a parcel.
Attention	attention_digit_span_forward	digit_span_forward	0.35	I will read a sequence of numbers. Please repeat them in the same order: 2, 1, 8, 5, 4.
Attention	attention_digit_span_forward_alt	digit_span_forward	0.40	Please repeat the following numbers in the same order: 3, 9, 2, 7, 6, 5.
Attention	attention_serial3	serial3	0.45	Starting from 50, subtract 3 each time and say six numbers in a row.
Attention	attention_digit_span_backward	digit_span_backward	0.50	Now repeat these numbers backward: 7, 4, 2.
Attention	att_math_simple_1	math_word_problem	0.50	Please calculate mentally: if you have 3 apples, receive 4 more, and then eat 2 of them, how many apples are left?
Attention	att_math_money_1	math_word_problem	0.55	A book costs 8 yuan. If you pay 20 yuan, how much change should you receive?
Attention	att_number	circle_number_naming	0.55	Name all the numbers inside the circles in the picture.
Attention	digit_span_backward_alt	backward	0.60	Try again: please repeat 9, 5, 1, 3 backward.
Attention	att_math_multi_1	math_word_problem	0.60	If you buy 3 bags of candy and each bag costs 6 yuan, how much do they cost in total?
Attention	att_seq_digits_1	digit_sequencing	0.65	I will read several numbers. Please say them in ascending order: 5, 1, 3.
Attention	attention_serial7	serial7	0.70	Starting from 100, subtract 7 each time and say five numbers in a row.
Attention	att_math_time_1	math_word_problem	0.70	If a person walks 4 kilometers per hour, how many hours are needed to walk 20 kilometers?
Attention	att_seq_digits_2	digit_sequencing	0.75	Please arrange these numbers in ascending order: 8, 2, 9, 5.
Attention	att_math_fraction_1	math_word_problem	0.75	If you have 18 cookies and give one third of them to a friend, how many cookies do you give away?
Attention	att_math_complex_1	math_word_problem	0.80	An item originally costs 100 yuan and is sold at 20% off. What is the discounted price?
Attention	att_seq_digits_3	digit_sequencing	0.85	This time, arrange the numbers in descending order: 12, 6, 20.
Attention	att_seq_letter_num	digit_sequencing	0.90	First say the numbers in ascending order, then say the letters. For example, if I say A, 3, B, 1, you should say 1, 3, A, B. Now try this set: A, 5, B, 2.
Language	lang_info_capital	general_knowledge	0.20	What city is the capital of China?
Language	lang_info_sun	general_knowledge	0.20	In which direction does the sun rise?
Language	lang_info_season	general_knowledge	0.30	Which season comes after spring in a year?
Language	lang_comp_clothes	social_comprehension	0.30	Why do we wash clothes?
Language	lang_info_boil	general_knowledge	0.35	At what temperature does water boil?
Language	language_sentence_rep_1	sentence_repetition	0.45	Please repeat this sentence completely: I only know that Mr. Zhang came to help today.
Language	language_sentence_rep_3	sentence_repetition	0.50	Please repeat this sentence: If the weather is suitable, we will go to the park for a walk tomorrow.
Language	lang_comp_wallet	social_comprehension	0.50	If you find a wallet on the road containing an ID card and money, what should you do?
Language	lang_comp_traffic	social_comprehension	0.55	Why should we obey traffic rules?
Language	language_sentence_rep_2	sentence_repetition	0.55	Please repeat: When the dog is in the backyard, the cat always hides under the sofa.

Continued on next page

Domain	Question ID	Type	Diff.	Instruction
Language	lang_info_geo	general_knowledge	0.60	On which continent is Italy located?
Language	lang_comp_food	social_comprehension	0.60	Why do food packages now print the production date?
Language	lang_info_liter	general_knowledge	0.65	Who is the author of Dream of the Red Chamber?
Language	verbal_fluency_animals	verbal_fluency	0.70	Name as many different animals as possible.
Language	lang_info_science	general_knowledge	0.75	What usually causes iron to rust? In other words, what does iron come into contact with when it rusts?
Language	lang_comp_license	social_comprehension	0.85	Why must doctors and lawyers have licenses before they can practice?
Abstraction	abs_wais_sim_coat_suit	abstraction_pairs	0.35	What do a coat and a suit have in common?
Abstraction	abs_wais_sim_table_chair	abstraction_pairs	0.40	What do a table and a chair have in common?
Abstraction	abs_wais_sim_eye_ear	abstraction_pairs	0.45	What do eyes and ears have in common?
Abstraction	abstraction_pairs_alt1	abstraction_pairs	0.45	What do a television and a radio have in common?
Abstraction	abstraction_pairs	abstraction_pairs	0.50	What do a banana and an orange have in common?
Abstraction	abstraction_pairs_new	abstraction_pairs	0.55	What do a mobile phone and a letter have in common?
Abstraction	abs_wais_sim_air_water	abstraction_pairs	0.60	What do air and water have in common?
Abstraction	abstraction_pairs_alt2	abstraction_pairs	0.60	What do shoes and hats have in common?
Abstraction	abs_wais_sim_poem_statue	abstraction_pairs	0.70	What do a poem and a statue have in common?
Abstraction	abs_wais_sim_fly_tree	abstraction_pairs	0.75	What do a fly and a tree have in common?
Abstraction	abs_wais_sim_praise_punish	abstraction_pairs	0.85	What do praise and punishment have in common?
Abstraction	abs_wais_sim_enemy_friend	abstraction_pairs	0.85	What do an enemy and a friend have in common?
Abstraction	abs_wais_sim_wish_reality	abstraction_pairs	0.90	What do wishes and reality have in common?
Abstraction	abs_wais_sim_law_freedom	abstraction_pairs	0.95	What do the law and freedom have in common?
Delayed Recall	delayed_recall_5words	delayed_recall_5words	0.60	Earlier I asked you to remember five words. Please now say as many of those words as you can still remember.
Orientation	orientation_place	orientation_place	0.40	Please remember this location statement: This is the memory clinic of Sunshine Community Health Service Center. Now repeat where we are.
Orientation	orientation_time_common	orientation_time_common	0.50	In a year, which season does the Spring Festival usually fall in? What about the Mid-Autumn Festival?
Orientation	ori_city_province	orientation_place_detail	0.40	Please remember this location label: Wuhou District, Chengdu City, Sichuan Province. Now say the province, city, and district in order.
Orientation	orientation_personal	orientation_personal	0.45	Please remember this identity card: Aunt Wang, 62 years old, born in 1964, retired teacher. Now say her age and year of birth.
Orientation	ori_place_floor	orientation_place_detail	0.55	Please remember this location statement: We are now on the 3rd floor of the rehabilitation building, in the neurology outpatient clinic. Please say which floor and which department.
Orientation	orientation_time	orientation_time	0.60	Please remember this information: Today is Thursday, April 23, 2026. What day of the week will it be three days later? What date will it be one week later?
Orientation	ori_time_season	orientation_time_detail	0.55	Please remember: October 3, 2026. Is this in autumn? If yes, which months does autumn usually include?

Table 11: Complete question bank used in our experiments.

E.2 Compact Q-matrix Table

Question ID	Domain	Top-1	Top-2	Top-3
vis_analog_1	Vis./Exec.	Abstraction (0.50)	Language (0.20)	Vis./Exec. (0.10)
vis_analog_2	Vis./Exec.	Abstraction (0.50)	Language (0.20)	Vis./Exec. (0.10)
vis_logic_seq_1	Vis./Exec.	Vis./Exec. (0.50)	Attention (0.20)	Abstraction (0.20)
vis_logic_seq_2	Vis./Exec.	Vis./Exec. (0.60)	Attention (0.25)	Language (0.05)
vis_cube	Vis./Exec.	Vis./Exec. (0.85)	Naming (0.05)	Language (0.05)
vis_clock_8_20	Vis./Exec.	Vis./Exec. (0.75)	Language (0.10)	Orientation (0.10)
vis_logic_seq_3	Vis./Exec.	Vis./Exec. (0.60)	Attention (0.15)	Abstraction (0.15)
vis_trails	Vis./Exec.	Vis./Exec. (0.60)	Attention (0.30)	Language (0.05)
vis_clock_11_10	Vis./Exec.	Vis./Exec. (0.70)	Attention (0.10)	Language (0.10)
vis_analog_3	Vis./Exec.	Abstraction (0.67)	Vis./Exec. (0.11)	Language (0.11)
vis_logic_seq_4	Vis./Exec.	Vis./Exec. (0.50)	Attention (0.20)	Abstraction (0.20)
vis_analog_4	Vis./Exec.	Abstraction (0.40)	Language (0.20)	Vis./Exec. (0.10)
vis_map_route	Vis./Exec.	Vis./Exec. (0.70)	Language (0.15)	Attention (0.10)
vis_logic_seq_5	Vis./Exec.	Vis./Exec. (0.50)	Abstraction (0.20)	Attention (0.15)
vis_logic_seq_fib	Vis./Exec.	Vis./Exec. (0.50)	Abstraction (0.30)	Attention (0.10)
naming_def_1	Naming	Naming (0.70)	Language (0.10)	Vis./Exec. (0.05)
naming_body_parts	Naming	Naming (0.70)	Vis./Exec. (0.10)	Language (0.10)
naming_def_3	Naming	Naming (0.70)	Language (0.15)	Vis./Exec. (0.05)
naming_animals	Naming	Naming (0.80)	Language (0.06)	Vis./Exec. (0.05)
naming_def_2	Naming	Naming (0.70)	Language (0.15)	Vis./Exec. (0.05)
naming_objects	Naming	Naming (0.80)	Language (0.06)	Vis./Exec. (0.05)
naming_def_5	Naming	Naming (0.80)	Language (0.06)	Vis./Exec. (0.05)
naming_def_4	Naming	Naming (0.85)	Language (0.10)	Abstraction (0.05)
memory_learning_5words	Memory	Memory (0.80)	Attention (0.10)	Naming (0.05)
memory_short_story	Memory	Memory (0.60)	Language (0.15)	Attention (0.10)
attention_digit_span_forward	Attention	Attention (0.75)	Memory (0.10)	Vis./Exec. (0.05)
attention_digit_span_forward_alt	Attention	Attention (0.75)	Memory (0.10)	Vis./Exec. (0.05)
attention_serial3	Attention	Attention (0.70)	Memory (0.10)	Orientation (0.10)
attention_digit_span_backward	Attention	Attention (0.70)	Memory (0.10)	Vis./Exec. (0.05)
att_math_simple_1	Attention	Attention (0.60)	Memory (0.10)	Language (0.10)
att_math_money_1	Attention	Attention (0.60)	Language (0.10)	Orientation (0.10)
att_number	Attention	Attention (0.56)	Naming (0.17)	Vis./Exec. (0.11)
digit_span_backward_alt	Attention	Attention (0.70)	Memory (0.10)	Vis./Exec. (0.05)
att_math_multi_1	Attention	Attention (0.78)	Language (0.11)	Memory (0.06)
att_seq_digits_1	Attention	Attention (0.71)	Memory (0.12)	Vis./Exec. (0.06)
attention_serial7	Attention	Attention (0.80)	Vis./Exec. (0.05)	Memory (0.05)
att_math_time_1	Attention	Attention (0.67)	Language (0.11)	Abstraction (0.11)
att_seq_digits_2	Attention	Attention (0.78)	Vis./Exec. (0.11)	Memory (0.06)
att_math_fraction_1	Attention	Attention (0.60)	Language (0.10)	Abstraction (0.10)
att_math_complex_1	Attention	Attention (0.70)	Language (0.10)	Vis./Exec. (0.05)
att_seq_digits_3	Attention	Attention (0.60)	Memory (0.10)	Delayed Recall (0.10)
att_seq_letter_num	Attention	Attention (0.60)	Memory (0.10)	Language (0.10)
lang_info_capital	Language	Language (0.80)	Delayed Recall (0.10)	Orientation (0.10)
lang_info_sun	Language	Language (0.60)	Delayed Recall (0.10)	Vis./Exec. (0.05)
lang_info_season	Language	Language (0.80)	Orientation (0.20)	–
lang_comp_clothes	Language	Language (0.70)	Abstraction (0.20)	Orientation (0.10)
lang_info_boil	Language	Language (0.90)	Delayed Recall (0.10)	–
language_sentence_rep_1	Language	Language (0.50)	Attention (0.20)	Memory (0.15)
language_sentence_rep_3	Language	Language (0.40)	Attention (0.25)	Memory (0.20)
lang_comp_wallet	Language	Language (0.40)	Abstraction (0.20)	Delayed Recall (0.10)
lang_comp_traffic	Language	Language (0.50)	Abstraction (0.20)	Attention (0.08)
language_sentence_rep_2	Language	Language (0.56)	Attention (0.22)	Memory (0.17)
lang_info_geo	Language	Language (0.80)	Orientation (0.20)	–
lang_comp_food	Language	Language (0.60)	Abstraction (0.30)	Attention (0.05)
lang_info_liter	Language	Language (0.60)	Memory (0.10)	Delayed Recall (0.10)
verbal_fluency_animals	Language	Language (0.40)	Naming (0.30)	Attention (0.10)
lang_info_science	Language	Language (0.80)	Memory (0.05)	Attention (0.05)
lang_comp_license	Language	Abstraction (0.35)	Language (0.30)	Delayed Recall (0.10)
abs_wais_sim_coat_suit	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abs_wais_sim_table_chair	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abs_wais_sim_eye_ear	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abstraction_pairs_alt1	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abstraction_pairs	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abstraction_pairs_new	Abstraction	Abstraction (0.60)	Language (0.10)	Vis./Exec. (0.05)
abs_wais_sim_air_water	Abstraction	Abstraction (0.70)	Language (0.10)	Naming (0.05)
abstraction_pairs_alt2	Abstraction	Abstraction (0.60)	Language (0.10)	Vis./Exec. (0.05)
abs_wais_sim_poem_statue	Abstraction	Abstraction (0.60)	Language (0.10)	Vis./Exec. (0.05)
abs_wais_sim_fly_tree	Abstraction	Abstraction (0.78)	Language (0.11)	Naming (0.06)
abs_wais_sim_praise_punish	Abstraction	Abstraction (0.80)	Language (0.10)	Attention (0.05)
abs_wais_sim_enemy_friend	Abstraction	Abstraction (0.80)	Language (0.10)	Attention (0.05)
abs_wais_sim_wish_reality	Abstraction	Abstraction (0.80)	Language (0.10)	Attention (0.05)
abs_wais_sim_law_freedom	Abstraction	Abstraction (0.80)	Language (0.10)	Attention (0.05)
delayed_recall_5words	Delayed Recall	Delayed Recall (0.70)	Memory (0.15)	Naming (0.05)
orientation_place	Orientation	Orientation (0.60)	Memory (0.10)	Attention (0.10)
orientation_time_common	Orientation	Orientation (0.60)	Language (0.10)	Abstraction (0.10)
ori_city_province	Orientation	Orientation (0.70)	Memory (0.10)	Language (0.10)
orientation_personal	Orientation	Delayed Recall (0.60)	Memory (0.10)	Attention (0.10)
ori_place_floor	Orientation	Orientation (0.60)	Memory (0.10)	Attention (0.10)
orientation_time	Orientation	Orientation (0.40)	Attention (0.20)	Delayed Recall (0.15)
ori_time_season	Orientation	Orientation (0.70)	Attention (0.10)	Language (0.10)

Table 12: Compact Q-matrix summary. Each entry reports one of the three highest-weighted cognitive dimensions and its normalized Q-matrix weight. A dash indicates that no additional positive-weight dimension is assigned.

E.3 Prompt

E.3.1 Q-matrix Annotation Prompt

Q-matrix Annotation Prompt

```
SYSTEM PROMPT:
You are an experienced specialist in neurology, geriatric medicine, and cognitive assessment, as well as a rigorous data
      annotator. Your task is to read a cognitive assessment item and annotate continuous Q-matrix weights.

Rules:
1. Output JSON only, with no extra text.
2. Assign weights in [0, 1] to the eight cognitive dimensions. The weights must sum to 1.0.
3. Most items mainly measure 1--2 dimensions. Do not allocate weights uniformly across many dimensions.
4. If an item directly measures one dimension, that dimension should receive a dominant weight.
5. Small secondary weights are allowed, but they should be conservative.
6. The confidence score must be in [0, 1].
7. The reasoning field should briefly explain the core basis for the annotation.
8. The evidence field should list 2--4 key cues.

Dimensions:
- visuospatial_executive: visuospatial processing, planning, sequence switching, executive control, route planning, clock
      drawing, cube copying, and trail-making.
- naming: object, animal, and concept naming; lexical retrieval.
- memory_learning: immediate learning, encoding, and retention of new information.
- attention: sustained attention, working memory, digit span, serial calculation, sequencing, and dual processing.
- language: sentence repetition, language comprehension, verbal fluency, semantic expression, and common-knowledge language
      interaction.
- abstraction: superordinate concepts, analogical reasoning, and abstract relation understanding.
- delayed_recall: delayed recall and retrieval of previously learned information.
- orientation: orientation to time, place, person, and personal information.

USER PROMPT TEMPLATE:
Please annotate the following item with continuous Q-matrix weights.

Input fields:
question_id, domain, instruction, type, difficulty, negative_criteria, image, and extra metadata.

Output JSON schema:
{
  "question_id": "...",
  "weights": {
    "visuospatial_executive": 0.0,
    "naming": 0.0,
    "memory_learning": 0.0,
    "attention": 0.0,
    "language": 0.0,
    "abstraction": 0.0,
    "delayed_recall": 0.0,
    "orientation": 0.0
  },
  "confidence": 0.0,
  "reasoning": "short explanation",
  "evidence": ["string", "string"]
}
```

E.3.2 LLM-based Response Evaluation Prompt

Output Schema

```
The model should return a JSON object with the following fields:

{
  "severity": integer,   // one of {0, 1, 2}
  "confidence": float,   // a value in [0, 1]
  "reason": string,      // a short explanation of the judgment
  "evidence": string     // key evidence from the user's response
}

Severity labels are defined as:
0: normal performance
1: mild or suspicious abnormality
2: clear abnormality
```

System Prompt for Response Evaluation

```
You are a clinical cognitive assessment assistant.

Your task is to read the question instruction, the negative criterion, and the user's answer. Then evaluate the answer by
```

```
      assigning a severity label and a confidence score.

Severity must be one of:
0 = normal performance
1 = mild or suspicious abnormality
2 = clear abnormality

Confidence must be a floating-point value between 0.0 and 1.0.

Use the following principles:
- If the user's answer clearly satisfies the requirement without errors or omissions, assign severity = 0.
- If the answer is partially complete, uncertain, or mildly incorrect, assign severity = 1.
- If the answer seriously violates the requirement or the user can barely complete the task, assign severity = 2.
- If the answer expresses uncertainty, such as "not sure", "maybe", or "I guess", assign a lower confidence score.
- If the answer is explicit and matches the scoring criterion, assign a higher confidence score.

Strictly follow the provided negative criterion when judging severity.

Return only a JSON object with the following fields:
{
  "severity": ...,
  "confidence": ...,
  "reason": "...",
  "evidence": "..."
}
```

User Prompt Template

```
instruction:
{question instruction}

negative_criteria:
{question-specific scoring criterion}

user_answer:
{free-form user response}
```