

Ability-Residual Decoupled Modeling for Affective Cognitive Diagnosis

Boyuan Zhao

Shaanxi Normal University

Meng Ye

yemeng@snnu.edu.cn

Shaanxi Normal University

Research Article

Keywords: cognitive diagnosis, affective cognitive diagnosis, ability residual, affect contamination, concept attention, intelligent education

Posted Date: October 28th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11300763/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Ability-Residual Decoupled Modeling for Affective Cognitive Diagnosis

Boyuan Zhao¹ and Meng Ye^{1*}

^{1*}Key Laboratory of Modern Teaching Technology, Shaanxi Normal University, 199 Chang'an South Road, Xi'an, 710062, Shaanxi, China.

*Corresponding author(s). E-mail(s): yemeng@snnu.edu.cn;
Contributing authors: 20243259@snnu.edu.cn;

Abstract

Cognitive diagnosis infers students' concept mastery from response logs. However, students' responses are not determined by mastery alone: non-cognitive factors such as emotion, engagement, and fatigue can also affect performance. Affective cognitive diagnosis therefore extends conventional cognitive diagnosis by incorporating affective states. Existing methods often assume that the cognitive diagnosis backbone has already explained ability, item, and concept effects, so the remaining errors can be attributed mainly to affect. We argue that this assumption can be insufficient in real educational data: item calibration bias, systematic concept bias, personalized student-concept deviations, and latent student-item matching can form stable cognitive residuals. Without an explicit modeling pathway, these residuals may leak into affective representations, producing affect contamination. To address this problem, we propose an ability-residual decoupled framework for affective cognitive diagnosis. The model first captures unmodeled cognitive residuals through student, item, concept, student-concept, and low-rank student-item components, and then uses an affective module to modulate guess/slip effects. A Q-matrix-constrained concept residual attention mechanism adaptively aggregates only item-relevant concept residuals. Experiments on ASSIST2017, ASSIST2012, ASSIST2009, and Junyi with six cognitive diagnosis backbones show response-prediction gains across the reported comparisons and generally improved affect alignment when affect labels are available. Ablation studies, leakage probes, principal component analysis visualization, long-tail analysis, and case studies further indicate that ability residuals absorb stable cognitive bias, reduce cognitive contamination in the affective branch, and enhance the robustness and predictive accuracy of cognitive diagnosis models. Our code is available at <https://github.com/psychosiwa/ARACD>.

Keywords: cognitive diagnosis, affective cognitive diagnosis, ability residual, affect contamination, concept attention, intelligent education

1 Introduction

Cognitive diagnosis (CD) infers students’ mastery of fine-grained knowledge concepts, skills, or cognitive attributes from their responses in exercises, tests, or online learning platforms (Junker and Sijtsma 2001), (Leighton and Gierl 2007). Unlike evaluation methods that report an overall score or a holistic ability estimate, CD emphasizes interpretable diagnostic results, i.e., which concepts a student has mastered and which remain deficient. Therefore, CD can support personalized exercise recommendation, learning path planning, formative assessment, and adaptive testing in intelligent tutoring systems (Leighton and Gierl 2007; VanLehn 2011; Feng et al. 2009). As student response logs continue to accumulate in large-scale online education, CD has emerged as a critical technology connecting educational measurement, student modeling, and intelligent educational decision-making (Wang et al. 2020; Piech et al. 2015; Su et al. 2024).

Research on cognitive diagnosis models (CDMs) has evolved from psychometric models to neural diagnostic models. Classic approaches include DINA, IRT, and MIRT. Specifically, DINA uses the Q-matrix to characterize the concepts required by an item and explains response noise through discrete mastery patterns and item parameters, such as guessing and slipping (Junker and Sijtsma 2001), (de la Torre 2009). Accordingly, IRT estimates response probabilities from the matching relationship between continuous latent ability and item difficulty (Lord 1980), while MIRT extends ability to a multidimensional latent space for multi-skill assessment tasks (Reckase 2009). Deep learning has recently been introduced into CD, affording models that can represent more complex interactions among students, items, and cognitive attributes. For instance, NCDM strengthens student-item-concept interaction modeling with neural networks while retaining interpretability (Wang et al. 2020). Alternative approaches, such as RCD and SCD, improve diagnosis under sparse-data conditions by using relational structures and self-supervised signals, respectively (Gao et al. 2021), (Wang et al. 2023). Higher-order ability modeling, global-local cognitive modeling, and attention mechanisms have also been studied to feature complex ability structures, learning dynamics, and item-related concept relations (Su et al. 2024), (Huang et al. 2025).

Although CDMs are designed to explain response behavior through cognitive ability or mastery, an observed response may be driven by latent cognitive factors, latent affective factors, or their joint influence (Wang et al. 2024). Existing studies report distinct roles for different affective factors, with control-value theory explaining how achievement emotions shape motivation, learning strategies, and performance (Pekrun 2006). Empirical work reveals that affective states evolve dynamically during complex learning (D’Mello and Graesser 2012), and that boredom and frustration differ in their incidence, persistence, and consequences across learning environments (Baker

et al. 2010). Furthermore, affect-detection research has introduced methods for estimating learner states (Calvo and D’Mello 2010), whereas affect-aware tutoring studies use these estimates to adapt learner-system interactions and feedback (Woolf et al. 2009; Conati and Maclaren 2009; D’Mello et al. 2007). Together, these findings lay the path for incorporating affective factors into cognitive diagnosis. Consequently, researchers have introduced affective states into cognitive diagnosis frameworks, leading to a new perspective, affective cognitive diagnosis. These models estimate students’ non-cognitive states through an affective module and use them to modulate guess/slip effects, thereby explaining response fluctuations among students with comparable knowledge mastery (Wang et al. 2024). Notably, when real affect labels are unavailable, these methods can draw on supervised or instance-level contrastive learning and use response labels to construct weak positive and negative relations for learning response-derived affective representations (Khosla et al. 2020), (Chen et al. 2020).

Existing affective cognitive diagnosis methods implicitly assume that the base cognitive diagnosis backbone has already explained student, item, and concept-related cognitive structure sufficiently, so that the remaining response variation mainly reflects affective states (Wang et al. 2020), (Gao et al. 2021), (Wang et al. 2023), (Wang et al. 2024). Nevertheless, this assumption can be insufficient in real educational data, as items may contain difficulty calibration bias; concepts may exhibit systematic prediction bias; a student’s mastery of a specific concept may deviate from his or her overall ability; and a student-item pair may involve personalized matching effects due to word-familiarity, problem-solving strategies, or hidden skill requirements (Su et al. 2024), (Koren et al. 2009), (Aggarwal 2016). When these stable cognitive residuals are not explicitly modeled, the affective module may be forced to absorb them. Consequently, the learned affective representation may encode cognitive-residual information associated with items, concepts, and student ability in addition to non-cognitive states. This phenomenon is defined as affect contamination, namely cognitive residual contamination in affective representations. Such contamination weakens interpretability and may force the model to misattribute cognitive residual variation to affective factors, thereby reducing diagnostic reliability (Calvo and D’Mello 2010), (Wang et al. 2024).

Motivated by this problem, this study develops an ability-residual decoupling framework for affective cognitive diagnosis. It instantiates it as ARACD or ARCACD depending on whether real affect supervision is available. Inspired by bias decomposition and low-rank interaction modeling in recommender systems (Koren et al. 2009; Aggarwal 2016), the proposed approach uses multiple structured bias terms to express unmodeled cognitive residual variation, including item, student, concept-shared, and student-concept personalized biases. Furthermore, we introduce a low-rank student-item interaction term to capture latent personalized matching effects without parameterizing a complete student-item interaction matrix. Concept-level residuals are aggregated by a Q-matrix-constrained attention mechanism that assigns student-item-pair-conditioned weights to item-related concept-shared and student-concept residuals. The resulting ability residual first corrects the base prediction on the logit scale, after which the affective module performs guess/slip modulation. This ordered design places stable cognitive residual correction before affective modulation, assigning cognitive residual and affective effects to separate modeling pathways (Wang et al. 2020;

Huang et al. 2025). Figure 1 illustrates the affect contamination and ability-residual decoupling: stable cognitive residuals not explained by the backbone may leak into the affective branch, whereas a dedicated ability-residual pathway provides a more appropriate structural destination for this residual information.

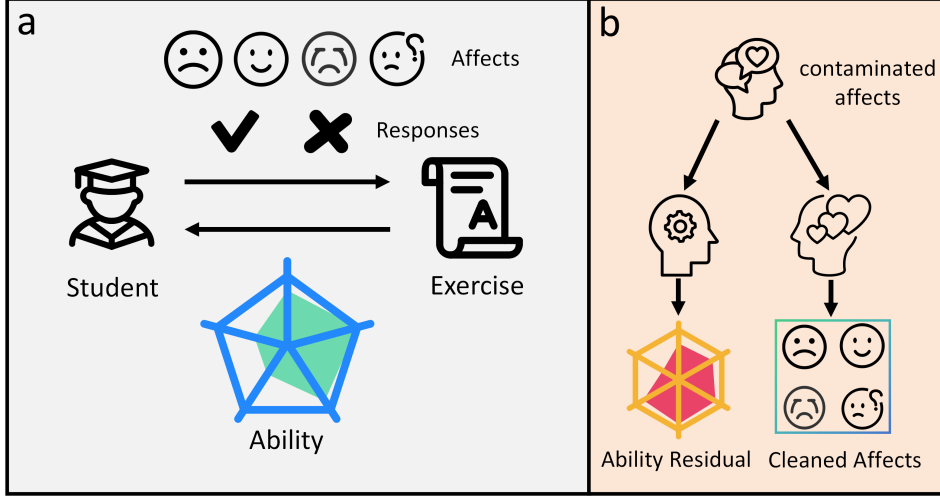


Fig. 1 Schematic illustration of affect contamination and ability-residual decoupling. Panel (a) illustrates that a student’s observed response to an exercise may reflect both cognitive ability and affective state. Panel (b) illustrates the proposed allocation of modeling roles: stable cognitive residual variation is assigned to the ability-residual pathway, reducing its leakage into the affective branch and allowing the affective representation to focus on non-cognitive modulation.

The main contributions of this paper are as follows.

1. We identify the affect contamination problem in affective cognitive diagnosis based on the observation that affective representations may absorb unmodeled cognitive residuals and become entangled with item, concept, and ability bias. This work complements existing affect-aware cognitive diagnosis studies, which focus on prediction improvement while paying less attention to representation contamination (Calvo and D’Mello 2010), (Wang et al. 2024).
2. We propose an ability-residual decoupling architecture that separates item, student, concept, student-concept personalized, and low-rank student-item interaction residuals from affective modeling. This strategy allows the affective module to focus more on non-cognitive factors.
3. We construct a Q-matrix-masked concept residual attention mechanism, which adaptively aggregates concept-shared residuals and student-concept personalized residuals as interpretable concept values within the item-related concept set. This approach improves the accuracy and interpretability of residual modeling.
4. We validate the proposed framework on ASSIST2017, ASSIST2012, ASSIST2009, and Junyi using multiple cognitive diagnosis backbones, including DINA, IRT,

MIRT, NCDM, RCD, and SCD. Our method’s effectiveness is systematically evaluated on response prediction, affect prediction, and representation decoupling through ablation experiments, residual group analysis, affect contamination diagnostics, and representation visualization (Junker and Sijtsma 2001), (Wang et al. 2020), (Lord 1980; Reckase 2009; Gao et al. 2021; Wang et al. 2023), (Wang et al. 2024).

2 Related Work

2.1 Cognitive Diagnosis and Affect-Aware Extensions

CD models estimate students’ mastery of fine-grained knowledge concepts or cognitive attributes from response logs, supporting personalized instruction, formative assessment, and adaptive testing (Leighton and Gierl 2007; VanLehn 2011; Feng et al. 2009). Early psychometric and cognitive diagnosis models have highly interpretable structures. Classic CDMs include DINA, IRT, and MIRT. Specifically, DINA relies on the Q-matrix to represent discrete item-concept requirements and uses item parameters such as guessing and slipping to explain response noise beyond mastery states (Junker and Sijtsma 2001), (de la Torre 2009). Besides, IRT models response probability as a function of continuous student ability and item difficulty (Lord 1980), while MIRT extends unidimensional ability to a multidimensional latent vector for multi-skill or multi-concept assessment (Reckase 2009). These models preserve clear educational-measurement meanings among students, items, and concepts, although parametric assumptions can limit their expressive power.

As researchers have accumulated large-scale response data on online education platforms, deep learning has been incorporated into cognitive diagnosis, giving rise to neural CDMs. An NCDM combines response probabilities with neural networks while retaining interpretable variables and monotonicity constraints (Wang et al. 2020). In this case, RCD models structural relations among students, items, and concepts, whereas SCD introduces self-supervised signals to alleviate sparsity and label deficiency (Gao et al. 2021), (Wang et al. 2023). Higher-order and global-local models further use hierarchical structures, attention, or local concept relations (Su et al. 2024), (Huang et al. 2025). These developments strengthen the representation of cognitive structure through more expressive student-, item-, and concept-level interactions, but their primary modeling target remains cognitive ability or mastery.

Observed responses are also affected by non-cognitive factors. Indeed, achievement emotions, confusion, frustration, and boredom can influence motivation, engagement, and performance (Pekrun 2006; D’Mello and Graesser 2012; Baker et al. 2010). At the same time, affect-detection and affect-aware tutoring studies provide empirical and computational support for incorporating these factors into learner modeling (Calvo and D’Mello 2010; Woolf et al. 2009; Conati and Maclaren 2009; D’Mello et al. 2007). Affect-aware cognitive diagnosis extends conventional CDMs by estimating affective states and uses these estimations to modulate guessing and slipping (Wang et al. 2024). When external affect labels are unavailable, response-conditioned contrastive learning can provide weak supervision for learning response-derived affective representations (Khosla et al. 2020), (Chen et al. 2020).

Although conventional and affect-aware CDMs improve cognitive-structure modeling and non-cognitive modulation, respectively, existing affect-aware extensions assume that the cognitive backbone has already explained stable student-, item-, and concept-related variation sufficiently (Wang et al. 2020), (Gao et al. 2021), (Wang et al. 2023), (Wang et al. 2024). In real educational data, item calibration bias, concept-level systematic bias, personalized student-concept deviations, and latent student-item matching may remain (Su et al. 2024), (Koren et al. 2009), (Aggarwal 2016). Notably, if these residuals are not explicitly modeled, the affective branch may absorb them and act as a generic error compensator. In this case, response improvements alone cannot establish that the learned affective representation is aligned with non-cognitive states (Calvo and D’Mello 2010), (Wang et al. 2024). Motivated by this gap, this study assigns stable cognitive residual variation to a dedicated ability-residual pathway before affective modulation.

2.2 Representation Decoupling

Representation learning aims to capture factors that are useful for downstream tasks (Bengio et al. 2013). Interestingly, a representation that improves prediction is not necessarily aligned with the target factor. Indeed, disentangled representation learning reveals that learned factors depend on task objectives, supervision signals, and structural inductive biases. Without appropriate inductive biases, general learning objectives cannot guarantee that a model will recover the expected factors (Locatello et al. 2019). This limitation is especially important when multiple latent factors are statistically correlated, because a model may encode shortcut features or spurious correlations that are useful for prediction but do not correspond to the intended factor (Geirhos et al. 2020).

Representation decoupling overcomes this limitation by encouraging different latent components to carry different types of information. Depending on the task, decoupling can be introduced through architectural separation, factor-specific supervision, contrastive objectives, or adversarial constraints. Although these mechanisms do not guarantee complete independence, they reduce unnecessary information leakage and improve interpretability. Typical diagnostic tools are linear probes, which test whether a representation contains information about a factor that the model intends to separate (Alain and Bengio 2016). Although these do not enforce decoupling, they provide empirical evidence about whether the intended separation has been achieved. This connection motivates the leakage probes used in this work to evaluate cognitive information remaining in affective representations.

2.3 Residual Modeling

Residual learning is a general modeling concept widely used to let models learn the remaining function relative to a base mapping (He et al. 2016). Bias correction and low-rank interaction modeling are widely used in recommender systems and educational prediction tasks to explain stable individual differences and sparse interaction structures (Koren et al. 2009), (Aggarwal 2016), (Rendle 2010; Thai-Nghe et al. 2010; Vie and Kashima 2019). Matrix factorization recommender models typically model user

bias, item bias, and low-rank user-item interactions to explain personalized differences that the global mean and the main model cannot capture (Koren et al. 2009), (Aggarwal 2016), (Rendle 2010). This concept has a natural interpretation when transferred to educational data: student bias corresponds to a student’s overall response tendency or baseline ability offset, item bias corresponds to item difficulty calibration error, and low-rank student-item interactions correspond to wording familiarity, problem-solving strategy matching, hidden skill requirements, or other unobserved matching relations. Similarly, response prediction in knowledge tracing and online assessment systems involves sparse interactions among students, items, and exercise records. Therefore, it requires additional structures to characterize individual differences and historical response patterns (Feng et al. 2009), (Piech et al. 2015), (Thai-Nghe et al. 2010), (Vie and Kashima 2019).

From this decoupling perspective, residual modeling offers the structural mechanism for assigning stable cognitive variation to a dedicated pathway rather than to the affective representation. In affective cognitive diagnosis, ability residuals are not merely numerical corrections to response logits but allocate modeling roles. Thus, residual patterns that recur with students, items, concepts, or student-item combinations should be explained mainly by the ability-residual pathway. In contrast, the affective module effectively handles fluctuations associated with transient states, engagement, or affective changes. Accordingly, the ability-residual module includes student, item, concept-shared, student-concept personalized, and low-rank student-item components. Hence, residual modeling supports representation decoupling in addition to increasing model capacity.

3 Problem Formulation

Let $\mathcal{S}$ be the set of students, be $\mathcal{I}$ the set of items, and $\mathcal{K}$ the set of concepts. For a student-item interaction (s, i) , where $s \in \mathcal{S}$ and $i \in \mathcal{I}$, the concept association vector of item i is denoted by $\mathbf{q}_i \in \{0, 1\}^K$, and the response label is $y_{si} \in \{0, 1\}$. If affect labels or affect vectors are available in the dataset, they are denoted by $\mathbf{a}_{si}$.

An observed response may reflect a latent cognitive state $\mathbf{c}_{si}$, a latent affective state $\mathbf{a}_{si}$, or their joint influence (Wang et al. 2024). Let p_{si} denote the probability of a correct response:

$$p_{si} = P(y_{si} = 1 \mid \mathbf{c}_{si}, \mathbf{a}_{si}, i, \mathbf{q}_i) \quad (1)$$

A conventional cognitive diagnosis model learns the following response probability:

$$\hat{y}_{si} = P(y_{si} = 1 \mid s, i, \mathbf{q}_i). \quad (2)$$

Affective cognitive diagnosis further introduces an affective representation $\mathbf{a}_{si}$, which is usually written as

$$\hat{y}_{si} = f_{\theta}(s, i, \mathbf{q}_i, \mathbf{a}_{si}). \quad (3)$$

However, when the base cognitive diagnosis backbone cannot sufficiently explain cognitive factors, the learned affective representation may contain the target non-cognitive state and the incorrectly absorbed cognitive residual, which is formulated as:

$$\mathbf{a}_{si} = \mathbf{a}_{si}^{affect} + \Delta_{si}^{cog}, \quad (4)$$

where $\mathbf{a}_{si}^{affect}$ is the non-cognitive component related to affective state and Δ_{si}^{cog} denotes cognitive residual variation that enters the affective space, without being explained by the base backbone, such as variation associated with item calibration bias, concept bias, or student-concept deviations.

Throughout this paper, *cognitive residuals* refer to stable cognitive variation not explained by the base cognitive diagnosis backbone. In contrast, the *ability residual* $r_{si}^{ability}$ is the structured, model-estimated correction introduced to capture such variation. *Ability residual* is used in a broad diagnostic sense and is not limited to a student-specific ability offset, incorporating item, concept, student-concept, and low-rank student-item components.

Besides, Δ_{si}^{cog} is a conceptual latent variation rather than an observed training target. By contrast, $r_{si}^{ability}$ is a structured correction estimated by the model from the available data and learning objectives. The two quantities are not equivalent, and response labels alone cannot uniquely identify their decomposition or recover either quantity as ground truth.

This paper aims to explicitly model the ability residual $r_{si}^{ability}$ so that the final response probability satisfies:

$$\hat{y}_{si} = f_{\theta}(s, i, \mathbf{q}_i, \mathbf{a}_{si}^{affect}, r_{si}^{ability}), \quad (5)$$

and to reduce the entanglement between $\mathbf{a}_{si}^{affect}$ and Δ_{si}^{cog} through a structured allocation of modeling roles. In other words, the ability residual module explains unmodeled cognitive residual variation, while the affective module explains how non-cognitive states modulate guessing, slipping, and response tendency.

4 Method

4.1 Overall Framework

The proposed framework has three core components: a base cognitive diagnosis backbone, an ability residual module, and an affective module. The base backbone F_{CD} outputs a base response probability according to student ability, item difficulty, item discrimination, and the Q-matrix:

$$y_{si}^{base} = F_{CD}(\mathbf{e}_s, \mathbf{d}_i, \mathbf{d}_i^{disc}, \mathbf{q}_i), \quad (6)$$

where $\mathbf{e}_s$ denotes student ability or mastery state, $\mathbf{d}_i$ is item difficulty or concept difficulty, $\mathbf{d}_i^{disc}$ denotes item discrimination, and $\mathbf{q}_i$ denotes the concepts involved in the item. DINA, IRT, MIRT, NCDM, RCD, or SCD can replace this backbone.

Then, the ability residual module introduces $r_{si}^{ability}$ to capture cognitive residual variation not explained by the base backbone and applies this correction on the logit scale:

$$z_{si}^{ability} = \text{logit}(y_{si}^{base}) + r_{si}^{ability}, \quad y_{si}^* = \sigma(z_{si}^{ability}). \quad (7)$$

Finally, the affective module generates an affective representation and modulates the residual-corrected response probability through the guess/slip mechanism. Unlike directly concatenating affective representations into the backbone, this structure explicitly distinguishes ability residuals from affective factors. Therefore, it is consistent with the modeling assumption on affect contamination presented in this paper.

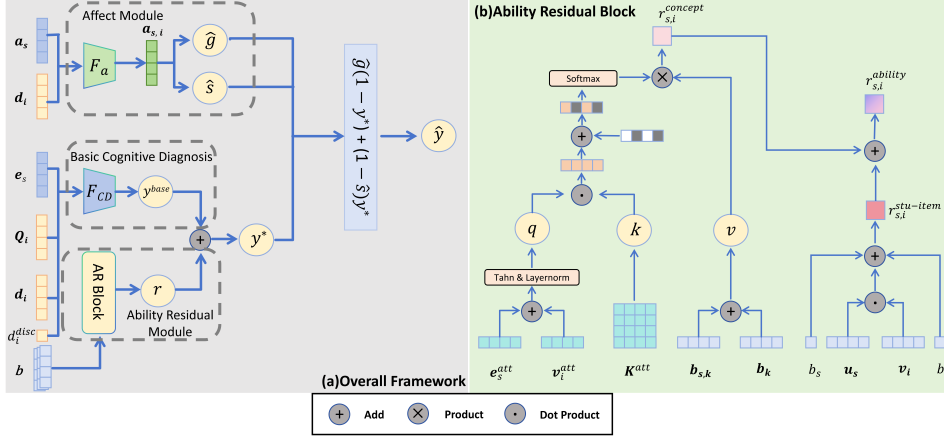


Fig. 2 Overall ARACD algorithmic framework and ability residual module.

4.2 Base Cognitive Diagnosis Backbone

The base cognitive diagnosis backbone is treated as a replaceable module. Considering NCDM as an example, the student knowledge mastery, item concept difficulty, and item discrimination vectors jointly form an interaction representation:

$$\mathbf{x}_{si} = \mathbf{d}_i^{disc} \odot (\mathbf{e}_s - \mathbf{d}_i^{kc}) \odot \mathbf{q}_i, \quad (8)$$

$$y_{si}^{base} = \sigma(MLP(\mathbf{x}_{si})). \quad (9)$$

where $\odot$ denotes element-wise multiplication, $\mathbf{d}_i^{disc}$ is item discrimination, and $\mathbf{d}_i^{kc}$ is the item's concept difficulty representation. For IRT and MIRT, the backbone formulation reduces to a matching function between unidimensional or multidimensional latent ability and item parameters. Accordingly, for DINA, the backbone estimates response probability based on whether the student has mastered all concepts required by the item and on guess/slip parameters. The proposed framework is designed as a plug-in that can be attached to different cognitive diagnosis models.

4.3 Ability Residual Modeling

The ability residual explains cognitive residual variation not captured by the base backbone. However, to improve modeling stability and interpretability, the ability

residual is divided into a student-item residual layer and a concept residual layer. The former layer models overall offsets and personalized interactions that do not depend on the concept structure, including student global bias, item global bias, and low-rank student-item latent interactions. The latter layer uses Q-matrix constraints to aggregate concept-shared residuals and student-concept personalized residuals within the concepts related to the item. This division separates non-concept-level response bias from concept-level knowledge bias, making the residual structure more compact and interpretable.

This modeling strategy absorbs cognitive residuals from non-affective sources because each residual term corresponds directly to a common stable error source in educational measurement. Specifically, item global bias absorbs item difficulty calibration errors; student global bias absorbs a student’s overall response tendency or baseline ability offset; concept-shared residuals absorb systematic prediction bias on certain concepts; student-concept personalized residuals absorb deviations in a student’s mastery of specific concepts; and low-rank student-item interactions absorb student-item-specific errors caused by wording familiarity, strategy matching, or hidden skill requirements. These factors commonly appear with the structure of students, items, or concepts, rather than being determined instantaneously by the affective state in a single interaction. Therefore, explicitly placing them in the ability residual pathway is a more appropriate modeling pathway for cognitive bias not explained by the base backbone.

During training, the ability residual corrects the base backbone output on the logit scale and is connected to the final response loss. If a certain type of error repeatedly appears for the same student, item, concept, or similar student-item combinations, gradients update the corresponding residual parameters and assign this stable error to the ability residual pathway. In contrast, the affective module modulates the final response probability through guess/slip and is more suitable for explaining response fluctuations induced by non-cognitive states. This structural division does not assume that the two are completely separable. Instead, it reduces the pressure on the affective branch to absorb cognitive bias through parameter allocation and educational structural constraints, thereby mitigating affect representation contamination.

$$r_{si}^{ability} = w_{SI}r_{si}^{stu-item} + w_Cr_{si}^{concept}, \quad (10)$$

where $r_{si}^{stu-item}$ is the student-item residual layer, $r_{si}^{concept}$ is the concept residual layer, and w_{SI} and w_C are the scaling coefficients of the two layers. Here w_{SI} is an overall weight for the student-item residual layer rather than a parameter that varies separately with sample (s, i) . In implementation, it can be set as either a learnable parameter or fixed to 1.

All b terms in the ability residual are internal learnable parameters rather than externally provided labels or manually calculated rules. During training, the model retrieves these residual terms from the corresponding parameter tables according to student, item, and concept IDs, and updates them end-to-end through the response prediction loss and affect representation learning loss. To avoid ambiguity in notation, the main symbols used in ability residual decomposition and concept attention are summarized below.

Table 1 Main notation used in the ability residual decomposition and concept residual attention.

Symbol	Meaning
s	The student in the considered student-item pair
i	The item in the considered student-item pair
k	The k -th concept
$\mathbf{q}_i$	Concept mask of item i
$\mathcal{K}_i$	The set of concepts involved in item i , $\mathcal{K}_i = \{k : q_{ik} = 1\}$
b_s	Stable offset indicating that student s is more or less likely than the overall level to answer correctly
b_i	Difficulty calibration error or item-specific systematic offset of item i
$b_k, \mathbf{b}$	Systematic residual shared by concept k across all students and items, and the vector of all concept-shared residuals
$b_{s,k}, \mathbf{b}_s^{concept}$	Personalized mastery deviation of student s on concept k relative to his or her overall ability, and the vector of all concept-specific personalized residuals for student s
$v_{s,k}, \mathbf{v}_s$	Composite concept residual value for student s on concept k , $v_{s,k} = b_k + b_{s,k}$
$\mathbf{u}_s, \mathbf{v}_i$	Latent student-item matching relation, whose inner product approximates the complete interaction residual matrix
γ_{SI}	Strength coefficient of the low-rank student-item interaction residual
$\mathbf{e}_s^{att}$	Student-specific embedding used to generate the student-item-pair-conditioned query
$\mathbf{v}_i^{att}$	Item-specific embedding used to generate the student-item-pair-conditioned query
$\mathbf{q}_{si}^{att}$	Student-item-pair-conditioned query, $\mathbf{q}_{si}^{att} = LN(\tanh(\mathbf{e}_s^{att} + \mathbf{v}_i^{att}))$
$\mathbf{k}_k^{att}, \mathbf{K}^{att}$	Attention key of the k -th concept and all concept keys
$e_{sik}, \mathbf{e}_{si}$	Concept attention scores before Q-mask application
$m_{ik}, \mathbf{m}_i$	Q-matrix mask that restricts attention to item-related concepts
$\alpha_{sik}, \alpha_{si}$	Q-masked concept attention weights, $\alpha_{sik} = \text{softmax}_k(e_{sik} + m_{ik})$
r_{si}^{att}	Output of the concept residual layer, $r_{si}^{att} = \sum_k \alpha_{sik} v_{s,k}$

These residual biases are usually initialized to 0 so that the model initially degenerates as much as possible to the base cognitive diagnosis backbone. Low-rank interaction vectors and attention vectors are initialized with small random values. As training proceeds, if a certain type of student, item, or concept exhibits stable errors under the base backbone, gradients push the corresponding residual parameters away from 0, absorbing this part of cognitive bias into the ability residual pathway.

4.3.1 Student-Item Residual Layer

The student-item residual layer is formulated as:

$$r_{si}^{stu-item} = r_s^{student} + r_i^{item} + r_{si}^{interaction}, \quad (11)$$

which absorbs overall offsets from students and items and models fine-grained latent student-item matching. Specifically,

$$r_s^{student} = b_s, \quad r_i^{item} = b_i, \quad r_{si}^{interaction} = \gamma_{SI} \frac{\langle \mathbf{u}_s, \mathbf{v}_i \rangle}{\sqrt{d}}. \quad (12)$$

where b_s and b_i correspond to the student global bias and item global bias in the table above, $\mathbf{u}_s$ and $\mathbf{v}_i$ are the low-rank interaction vectors of the student and item, respectively, d is the low-rank dimension, and γ_{SI} is the strength coefficient of the low-rank student-item interaction residual, corresponding to `student_item_rank_scale`. When γ_{SI} is small, the low-rank interaction term has a weak influence on the final residual. When γ_{SI} is large, the model makes stronger use of personalized student-item matching to correct predictions from the base backbone. The main experiments use $\gamma_{SI} = 1.5$ by default. To avoid the high parameter cost and overfitting risk of directly modeling the complete student-item interaction matrix, we use the low-rank decomposition $\mathbf{u}_s^\top \mathbf{v}_i$ to represent personalized residuals at the student-item level. This design reduces computational and storage overhead while imposing a low-rank constraint on the interaction structure, thereby improving generalization.

4.3.2 Concept Residual Layer

The concept residual layer is defined as:

$$r_{si}^{concept} = r_{si}^{att}, \quad (13)$$

and models systematic concept-level bias and student-concept personalized bias. This layer introduces the Q-matrix-constrained concept residual attention mechanism to aggregate item-related concepts through attention weighting, as detailed in Section 4.4. Specifically,

$$r_{si}^{att} = \sum_{k=1}^K \alpha_{sik} (b_k + b_{s,k}), \quad (14)$$

where b_k is the shared concept residual bias of the k -th concept, representing systematic bias of that concept across all students and items; $b_{s,k}$ is the personalized residual bias of student s on concept k , representing the student’s mastery deviation on that specific concept. Their sum provides the concept-layer value, which denotes the composite concept residual of the current student on that concept. $\mathcal{K}_i = \{k : q_{ik} = 1\}$ is the set of concepts involved in item i , and the Q-matrix mask zeroes out the attention weights of irrelevant concepts.

This decomposition has clear interpretability. Indeed, in the student-item residual layer, b_s and b_i absorb overall student and item offsets, respectively, and the low-rank interaction term $r_{si}^{interaction}$ reflect latent matching relations induced by item wording, problem-solving strategy, or hidden skill requirements. In the concept residual layer, $b_k + b_{s,k}$ explains systematic concept bias and student-concept personalized mastery deviations. This two-layer architecture facilitates ablation analysis and allows direct statistics on the contributions of different residual sources to the final response probability.

4.4 Q-Matrix-Constrained Concept Residual Attention

Next, we introduce a Q-matrix-constrained concept residual attention mechanism into the concept residual layer $r_{si}^{concept}$. Among the concepts associated with item i , the mechanism adaptively determines the relative contribution of each explicit concept residual to the concept-level correction for student s 's predicted response to item i . Specifically, a student-item-pair-conditioned query is constructed by combining a student-specific and an item-specific attention embedding. Its compatibility with each candidate concept key provides the basis for residual allocation: a higher compatibility score assigns a larger relative weight to the corresponding concept residual. The Q-matrix further restricts this allocation to concepts required by the item. It should be noted that attention is used only for concept-layer aggregation, not for the student-item residual layer.

The proposed module is a structured residual aggregation rather than a conventional hidden-state attention (Vaswani et al. 2017). It does not use attention to create a new hidden value. Still, it leaves each scalar $v_{s,k}$ untransformed and uses attention only to determine its relative allocation. The resulting correction remains directly decomposable into shared and student-specific concept residuals.

During implementation, we directly learn concept key vectors, whose role is equivalent to $\mathbf{k}_k^{att} = \mathbf{W}_k \mathbf{c}_k$ in the theoretical expression. Specifically, the model does not explicitly store $\mathbf{W}_k$ and $\mathbf{c}_k$ as two separate objects but combines their product into a learnable vector $\mathbf{k}_k^{att}$. We use the symbols in the table above and describe the computation for a student-item pair (s, i) .

First, the query for the student-item pair is constructed from the student and the item:

$$\mathbf{q}_{si}^{att} = LN(\tanh(\mathbf{e}_s^{att} + \mathbf{v}_i^{att})), \quad \mathbf{q}_{si}^{att} \in \mathbb{R}^{d_{att}}. \quad (15)$$

where $\mathbf{e}_s^{att}$ and $\mathbf{v}_i^{att}$ are obtained from student and item embedding tables, respectively, and both are vectors of length d_{att} . $\tanh(\cdot)$ denotes the hyperbolic tangent activation function, which compresses the combined student-side and item-side query information into a bounded nonlinear space. Moreover, $LN(\cdot)$ denotes Layer Normalization, which stabilizes the scale of the query vector and avoids unstable fluctuations in subsequent dot-product scores caused by excessively large vector norms. For each concept k , the model directly learns a key vector:

$$\mathbf{k}_k^{att} \in \mathbb{R}^{d_{att}}, \quad \mathbf{K}^{att} = [\mathbf{k}_1^{att}; \dots; \mathbf{k}_K^{att}] \in \mathbb{R}^{K \times d_{att}}. \quad (16)$$

Therefore, for a student-item pair (s, i) , the unmasked attention score on concept k is defined as:

$$e_{sik} = \frac{(\mathbf{q}_{si}^{att})^T \mathbf{k}_k^{att}}{\sqrt{d_{att}}}, \quad e_{sik} \in \mathbb{R}. \quad (17)$$

By concatenating the scores of all concepts, vector $\mathbf{e}_{si} \in \mathbb{R}^K$ is obtained. To ensure that attention is allocated only over item-related concepts, the item-level Q-matrix mask $\mathbf{m}_i \in \mathbb{R}^K$ is introduced:

$$m_{ik} = \begin{cases} 0, & q_{ik} = 1, \\ -\infty, & q_{ik} = 0, \end{cases} \quad (18)$$

where q_{ik} is the 0/1 mark of item i on concept k in the Q-matrix. The final attention weight is:

$$\alpha_{sik} = \text{softmax}_k(e_{sik} + m_{ik}), \quad \boldsymbol{\alpha}_{si} \in \mathbb{R}^K. \quad (19)$$

The concept residual value remains an interpretable scalar. For concept k , $v_{s,k}$ is a scalar and for all concepts it forms the vector $\mathbf{v}_s = [v_{s,1}, \dots, v_{s,K}] \in \mathbb{R}^K$. In the proposed solution, the shared concept residual vector $\mathbf{b} \in \mathbb{R}^K$ is added to the student-concept residual vector $\mathbf{b}_s^{\text{concept}} \in \mathbb{R}^K$:

$$v_{s,k} = b_k + b_{s,k}, \quad \mathbf{v}_s \in \mathbb{R}^K. \quad (20)$$

The Q-matrix defines the candidate set $\mathcal{K}_i$, whereas the query-key compatibility scores determine the relative allocation within that set. Thus, α_{sik} reflects student-item-pair-concept compatibility rather than directly the magnitude of $v_{s,k}$, which is used only after the weights are obtained. Therefore, it is interpreted as a residual-allocation coefficient, and not as a mastery probability or causal importance score.

Concept residual aggregation is a weighted sum along the concept dimension, K :

$$r_{si}^{\text{att}} = \sum_{k=1}^K \alpha_{sik} v_{s,k}, \quad r_{si}^{\text{att}} \in \mathbb{R}. \quad (21)$$

For a single-concept item, the masked softmax gives $\alpha_{sik^*} = 1$ and $r_{si}^{\text{att}} = v_{s,k^*}$, and adaptive allocation is activated only for multi-concept items.

During implementation, the shared concept term and the student-concept personalized term can be separately summarized:

$$r_{si}^{\text{shared}} = \sum_{k=1}^K \alpha_{sik} b_k, \quad r_{si}^{\text{student-concept}} = \sum_{k=1}^K \alpha_{sik} b_{s,k}. \quad (22)$$

Their sum gives the concept residual layer $r_{si}^{\text{concept}} = r_{si}^{\text{att}}$, which is then combined with the student-item residual layer $r_{si}^{\text{stu-item}}$ to form the final ability residual.

4.5 Affective Module

The affective module learns the non-cognitive state of a student in the current interaction. Given the student affective representation $\mathbf{a}_s$ and the item concept difficulty representation $\mathbf{d}_i^{kc}$, the interaction-level affective representation is formulated as:

$$\mathbf{a}_{si} = g_\theta(\mathbf{a}_s, \mathbf{d}_i^{kc}), \quad (23)$$

where $\mathbf{a}_s$ is a learnable student-level affective representation obtained by student ID lookup, $\mathbf{d}_i^{kc}$ is the item concept difficulty representation in the base cognitive diagnosis backbone and provides the knowledge requirement context of the current item, and $g_\theta(\cdot)$ is a learnable mapping function used to generate the affective representation of

the current student-item interaction. These parameters are learned end-to-end through the training objective, rather than manually specified by external affective rules.

Let guess and slip be directly determined by the affective representation:

$$g_{si} = \sigma(\mathbf{w}_g^T \mathbf{a}_{si} + b_g), \quad (24)$$

$$s_{si} = \sigma(\mathbf{w}_{slip}^T \mathbf{a}_{si} + b_{slip}). \quad (25)$$

where $\mathbf{w}_g$ and b_g are the linear weight and bias of the guess branch, and $\mathbf{w}_{slip}$ and b_{slip} are the linear weight and bias of the slip branch. The terms b_g and b_{slip} belong only to the intercepts of the affective modulation branch. They are not part of the student, item, or concept residuals in the ability residual module.

The final response probability is presented in Eq. (26) and indicates that when y_{si}^* is low, the guess term may increase the probability of a correct response; when y_{si}^* is high, the slip term may decrease the probability of a correct response. The final probability is therefore jointly determined by the residual-corrected prediction, the guess term, and the slip term.

$$\hat{y}_{si} = (1 - s_{si})y_{si}^* + g_{si}(1 - y_{si}^*). \quad (26)$$

4.6 Affective Representation Learning With and Without Affect Labels

Depending on whether real affect labels are available in the dataset, ACD follows two affect-aware training strategies. When the training data contain external affect annotations, supervised affect prediction is used. When real affect annotations are unavailable, contrastive affect representation learning is used by constructing positive and negative samples based on response labels. Let $\mathcal{B}$ be the current training batch, $\mathbf{a}_i^{gt} \in [0, 1]^{d_{aff}}$ be the real affect vector of sample i , and $\hat{\mathbf{a}}_i$ be the model’s affective representation. Given that the original ACD treats affective states as continuous values, the supervised affect perception module uses mean squared error:

$$\mathcal{L}_a = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \|\hat{\mathbf{a}}_i - \mathbf{a}_i^{gt}\|_2^2. \quad (27)$$

This loss corresponds to the affective perception loss in ACD and makes the affective representation induced by student affective traits and item difficulty close to the external affect annotations. In ARACD, $\hat{\mathbf{a}}_i$ still participates in final response prediction through guess/slip modulation. Besides, the ability residual has no independent supervision label, as its parameters are updated jointly by backpropagation through the response prediction path and the affect prediction path.

For CACD/ARCACD settings without real affect labels, this study follows the contrastive affect perception idea in the original ACD and replaces $\mathcal{L}_a$ with a contrastive loss. Its basic assumption is that student-item interactions with the same response outcome are more likely to correspond to similar affective states. In contrast, interactions with different response outcomes are more likely to correspond to different affective states. For any sample i in a batch, the positive and negative sample sets are constructed as:

$$\mathcal{P}_i = \{j \in \mathcal{B} : j \neq i, y_j = y_i\}, \quad \mathcal{N}_i = \{k \in \mathcal{B} : y_k \neq y_i\}. \quad (28)$$

where $y_i \in \{0, 1\}$ is the response label of sample i . Let $\text{sim}(\cdot, \cdot)$ denote cosine similarity between two affective vectors and τ be the temperature coefficient. Following CACD, the contrastive affective loss is defined as:

$$\mathcal{L}_{ca} = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \frac{1}{|\mathcal{P}_i|} \sum_{j \in \mathcal{P}_i} \log \frac{\exp(\text{sim}(\hat{\mathbf{a}}_i, \hat{\mathbf{a}}_j)/\tau)}{\sum_{k \in \mathcal{N}_i} \exp(\text{sim}(\hat{\mathbf{a}}_i, \hat{\mathbf{a}}_k)/\tau)}, \quad (29)$$

where

$$\mathcal{I} = \{i \in \mathcal{B} : |\mathcal{P}_i| > 0, |\mathcal{N}_i| > 0\} \quad (30)$$

is the set of valid anchors. This training strategy does not introduce real affect labels for individual samples. Instead, it uses weak positive and negative relations formed by response labels to constrain the affective representation space, thereby providing latent affective cues for guess/slip modulation.

4.7 Training Objective

The response prediction loss follows the binary cross-entropy used in ACD. For a batch $\mathcal{B}$, let $\hat{y}_i$ be the final predicted probability and y_i the true response label. The loss is formulated as:

$$\mathcal{L}_{CDM} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)]. \quad (31)$$

In supervised ACD/ARACD settings with real affect labels, response prediction loss and affect prediction loss are optimized jointly:

$$\mathcal{L}_{sup} = \mathcal{L}_{CDM} + \lambda_a \mathcal{L}_a, \quad (32)$$

where λ_a is the trade-off coefficient of the affect prediction loss. $\mathcal{L}_{CDM}$ constrains the final response probability, and $\mathcal{L}_a$ constrains the predicted affective vector to align with the external affect annotations.

In CACD/ARCACD settings without real affect labels, $\mathcal{L}_{ca}$ replaces the supervised affect MSE, and the training objective becomes:

$$\mathcal{L}_{unsup} = \mathcal{L}_{CDM} + \lambda_c \mathcal{L}_{ca}, \quad (33)$$

where λ_c is the trade-off coefficient of the contrastive affect perception loss. This setting does not compute $\mathcal{L}_a$. The affective representation influences response prediction through the guess/slip branch and receives weak supervision from response labels through $\mathcal{L}_{ca}$. Apart from the supervised affect loss or contrastive affect loss, this study does not introduce additional decorrelation losses or residual losses. Necessary control of parameter scale is handled by optimizer weight decay or parameter clipping, but is not reported as an independent training objective.

5 Experimental Setup

5.1 Research Questions

The study is organized around the following research questions.

RQ1: Can ability-residual decoupling stably improve response prediction performance? This work compares Base, ACD/CACD, and ARACD/ARCACD on ASSIST2017, ASSIST2012, ASSIST2009, and Junyi, and examines whether ACC, AUC, RMSE, and MAE have consistent improvements across datasets and backbones.

RQ2: Does the ability residual help affect alignment? ACD and ARACD are compared using Affect RMSE and Affect MAE to test whether explicitly modeling cognitive residuals reduces the explanatory pressure on the affective branch caused by item, concept, and ability bias.

RQ3: Which residual components account for the benefit of ability residuals? Hierarchical ablations are conducted among Full, Only stu-item residual, Only concept residual, and w/o ability residual to evaluate the contributions of the student-item residual layer, the concept residual layer, and their combination to response prediction and affect prediction.

RQ4: How do the attention dimension and low-rank dimension affect response prediction and affect alignment? The concept attention dimension d_{att} and the student-item low-rank dimension $rank$ are modified to examine the sensitivity of response metrics and affect metrics.

RQ5: Do the two residual layers capture response-related structure rather than random perturbations? The student-item residual and concept residual are visualized as a two-dimensional residual space to examine whether they form response-related patterns.

RQ6: Can ability residuals mitigate affect contamination and improve model reliability? Synthetic contamination probes, affect representation leakage diagnostics, and PCA visualization are used to analyze whether affective representations mix in cognitive bias information, and whether ability residuals can transfer this bias to the ability-residual pathway.

RQ7: Can the proposed framework improve long-tail student diagnosis? We further reanalyze low-activity students to examine whether ability residuals improve diagnosis for students with few interactions.

5.2 Datasets

The experiments use the educational datasets: ASSIST2017, ASSIST2012, ASSIST2009, and Junyi. All contain student response records, item IDs, concept annotations, and response correctness labels. ASSIST2017 and ASSIST2012 contain real affect labels and are used for supervised ACD/ARACD experiments. ASSIST2009 and Junyi do not use real affect supervision and are used for CACD/ARCACD experiments without affect labels. For each dataset, a fixed train/validation/test split ratio of 7:2:1 is used, while the data processing pipeline and evaluation protocol are consistent across all models.

Table 2
Statistics of the four educational datasets.

Dataset	Students	Items	Concepts
ASSIST2017	1709	3162	102
ASSIST2012	28998	47124	265
ASSIST2009	4163	17746	123
Junyi	70627	2737	722

5.3 Baseline Models and Compared Methods

Six representative models are used as base backbones: DINA, IRT, MIRT, NCDM, RCD, and SCD to examine whether the ability residual framework can be attached as a general plug-in to different cognitive diagnosis backbones. These six backbones cover discrete cognitive diagnosis, unidimensional and multidimensional IRT, neural cognitive diagnosis, relational structure modeling, and self-supervised cognitive diagnosis. Therefore, these models allow us to evaluate the stability of the proposed method under different modeling assumptions and expressive capacities. In addition to base cognitive diagnosis backbones, ACD and CACD are used as affect-aware comparison methods to distinguish between "introducing an affective module" and "further introducing ability residuals".

Table 3
Cognitive diagnosis backbones used in the experiments.

Backbone	Description
DINA	A discrete cognitive diagnosis model that assumes students need to master all concepts required by an item to answer it correctly with high probability, and models guessing and slipping through guess/slip parameters.
IRT	A unidimensional item response theory model that estimates response probability based on the matching relation between student ability and item parameters, serving as a classical psychometric baseline.
MIRT	A multidimensional item response theory model that extends student ability into a multidimensional latent vector to characterize ability differences in multi-skill or multi-concept assessments.
NCDM	A neural cognitive diagnosis model that uses neural networks to fit response probabilities based on interpretable variables such as student mastery, item concept difficulty, and item discrimination.
RCD	A relation-aware cognitive diagnosis model that enhances interaction representations through relation structures among students, items, and concepts, and is used to evaluate the generalization of ability residuals on structured backbones.
SCD	A self-supervised cognitive diagnosis model that introduces self-supervised signals into diagnostic representations to improve robustness under sparse response scenarios.

Table 4
Affect-aware comparison methods.

Method	Description
ACD	A supervised affective cognitive diagnosis model that adds an affect-aware module to the base cognitive diagnosis backbone, uses real affect labels to constrain affective representations, and modulates response probability through guess/slip.
CACD	A contrastive affective cognitive diagnosis model for settings without real affect labels; it uses response labels to construct positive and negative sample relations for learning response-derived affective representations and modulates response prediction through these representations.

On ASSIST2017 and ASSIST2012, where real affect labels are available, we compare three methods for each backbone: Base uses only the base cognitive diagnosis backbone without an affective module or ability residual; ACD adds an affect-aware module to the same backbone and performs supervised affect prediction with real affect labels, but does not explicitly model ability residuals; ARACD adds the ability residual module and Q-matrix-constrained concept residual attention on top of ACD. This setting tests whether ability residuals can further improve response prediction and affect prediction beyond affect supervision.

On ASSIST2009 and Junyi, where real affect labels are not used, we compare Base, CACD, and ARCACD. CACD does not use per-sample real affect labels but constructs positive and negative sample relations from response labels for contrastive affective representation learning. ARCACD further introduces the ability residual module into CACD’s contrastive affective representation learning framework. This setting tests whether ability residuals can still combine with response-derived affective representations and improve response prediction when external affect annotations are absent.

5.4 Evaluation Metrics and Settings

The main experiments use two types of metrics: response prediction metrics, including ACC, AUC, RMSE, and MAE, and affect alignment metrics, used only in supervised ACD/ARACD settings with real affect labels, including Affect RMSE and Affect MAE. For the former type, ACC denotes prediction accuracy after thresholding at 0.5; AUC measures whether the model assigns higher probabilities to correct responses; and RMSE and MAE measure probability error against the binary response label, with lower values being better.

All metrics measure the error between the predicted affective vectors and real affect labels, and all methods use the same data preprocessing, train/validation/test split, and evaluation scripts to ensure comparability. ASSIST2017 and ASSIST2012 use supervised affective learning, optimizing response prediction loss and real affect prediction loss. Additionally, ASSIST2009 and Junyi do not use real affect labels and adopt response-label-based contrastive affective representation learning. Except for hyperparameter sensitivity experiments, the main experiments keep the same training configuration within each dataset and backbone, select model checkpoints according to validation performance, and report final metrics on the test set. The experimental

setup comprises a single NVIDIA GeForce RTX 3070 Ti GPU with 8 GB of memory. For model structural parameters, the main experiments use a concept attention dimension $d_{att} = 16$, a student-item low-rank interaction dimension $rank = 1024$, and a low-rank student-item residual strength coefficient $\gamma_{SI} = 1.5$ by default. All models are optimized with Adam using a learning rate of 5×10^{-4} . The supervised experiments on ASSIST2017 and ASSIST2012 use a batch size of 512, train for at most 35 epochs with early-stopping patience values of 5 and 2, respectively, and optimize $\mathcal{L}_{CDM} + \mathcal{L}_a$. The contrastive experiments on ASSIST2009 and Junyi optimize $\mathcal{L}_{CDM} + \mathcal{L}_{ca}$, use a batch size of 256 and patience 5, and train for at most 20 and 30 epochs, respectively; RCD and SCD on Junyi use a batch size of 64 because of memory demand.

6 Experimental Results and Analysis

This section reports response prediction, affect alignment, ablation, hyperparameter sensitivity, residual maps, contamination diagnostics, long-tail results, and a student-level case study according to the research questions. Sections 6.1–6.7 correspond to RQ1–RQ7, respectively, while Section 6.8 provides supplementary student-level evidence.

6.1 Response Prediction Results (RQ1)

This section answers RQ1, namely whether ability-residual decoupling can stably improve response prediction performance under multi-dataset and multi-backbone settings. Table 5 reports the supervised affective learning results on ASSIST2017 and ASSIST2012. For each backbone, Base, ACD, and ARACD are retained, and four response metrics are reported: ACC, AUC, RMSE, and MAE. Table 6 lists the results without real affect labels on ASSIST2009 and Junyi, comparing Base, CACD, and ARCACD to test whether ability residuals can combine with response-label contrastive affective representation learning and bring stable gains.

Table 5 highlights that ARACD provides consistently improved response prediction across the two supervised datasets. Compared with Base, ARACD improves ACC and AUC and reduces RMSE and MAE for every reported backbone-dataset pair. Compared with ACD, ARACD yields higher ACC and AUC and lower RMSE and MAE in all 12 comparisons. These patterns suggest that ability-residual modeling can complement supervised affect modeling by accounting for part of the remaining cognitive residual variation. However, the magnitude of the gain still depends on the backbone and dataset.

We further run CACD and ARCACD on ASSIST2009 and Junyi under the same six-backbone setting to supplement cross-dataset evidence. Table 6 reports ACC, AUC, RMSE, and MAE for the two datasets side by side. For ASSIST2009, Base results for each backbone are included as response prediction controls without affective modules or ability residuals.

Table 6 reveals that on ASSIST2009, ARCACD achieves higher ACC/AUC and lower RMSE/MAE than both Base and CACD on all six backbones. In particular, NCDM+ARCACD reaches ACC/AUC of 0.7884/0.8487 under the targeted configuration with fixed $d_{att} = 16$, $rank = 1024$. In supplementary runs that combine ability

Table 5 Response prediction results on ASSIST2017 and ASSIST2012.

Backbone	Method	ASSIST2017				ASSIST2012			
		ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$
DINA	Base	0.6471	0.6950	0.4681	0.4383	0.7118	0.6923	0.4401	0.3654
	ACD	0.7103	0.7760	0.4374	0.4008	0.7352	0.7397	0.4234	0.3489
	ARACD	0.7390	0.8079	0.4202	0.3449	0.7494	0.7609	0.4139	0.3340
IRT	Base	0.6582	0.7224	0.4668	0.4086	0.7286	0.7242	0.4289	0.3440
	ACD	0.7152	0.7843	0.4338	0.3732	0.7401	0.7518	0.4194	0.3354
	ARACD	0.7394	0.8082	0.4201	0.3442	0.7489	0.7602	0.4142	0.3334
MIRT	Base	0.6801	0.7399	0.4661	0.4321	0.7355	0.7232	0.4467	0.3788
	ACD	0.7232	0.7935	0.4289	0.3821	0.7413	0.7538	0.4186	0.3453
	ARACD	0.7391	0.8078	0.4202	0.3455	0.7490	0.7606	0.4140	0.3341
NCDM	Base	0.6908	0.7520	0.4527	0.4191	0.7397	0.7451	0.4218	0.3344
	ACD	0.7230	0.7938	0.4296	0.4018	0.7421	0.7502	0.4185	0.3336
	ARACD	0.7394	0.8078	0.4203	0.3461	0.7487	0.7617	0.4138	0.3316
RCD	Base	0.7141	0.7795	0.4354	0.3713	0.7401	0.7477	0.4214	0.3373
	ACD	0.7219	0.7915	0.4296	0.3645	0.7445	0.7570	0.4175	0.3381
	ARACD	0.7364	0.8056	0.4210	0.3576	0.7499	0.7611	0.4131	0.3328
SCD	Base	0.7146	0.7802	0.4350	0.3676	0.7415	0.7549	0.4188	0.3371
	ACD	0.7259	0.7930	0.4289	0.3608	0.7460	0.7613	0.4153	0.3305
	ARACD	0.7357	0.8055	0.4212	0.3598	0.7497	0.7618	0.4132	0.3301

Table 6 Response prediction results on ASSIST2009 and Junyi.

Backbone	Method	ASSIST2009				Junyi			
		ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$
DINA	Base	0.7015	0.7206	0.4590	0.4473	0.7281	0.7115	0.4334	0.3866
	CACD	0.7240	0.7513	0.4344	0.3935	0.7456	0.7539	0.4100	0.3575
	ARCACD	0.7786	0.8396	0.3896	0.2799	0.7658	0.7582	0.4013	0.3298
IRT	Base	0.7175	0.7354	0.4391	0.3979	0.7335	0.7207	0.4113	0.3554
	CACD	0.7255	0.7571	0.4289	0.3581	0.7686	0.7583	0.4009	0.3284
	ARCACD	0.7819	0.8428	0.3858	0.2887	0.7692	0.7591	0.4001	0.3251
MIRT	Base	0.6894	0.6986	0.4574	0.4414	0.7355	0.7243	0.4215	0.3877
	CACD	0.7242	0.7499	0.4313	0.3810	0.7580	0.7537	0.4050	0.3436
	ARCACD	0.7807	0.8415	0.3882	0.2786	0.7683	0.7597	0.4004	0.3271
NCDM	Base	0.7188	0.7513	0.4358	0.3921	0.7262	0.7162	0.4453	0.4001
	CACD	0.7253	0.7573	0.4332	0.3845	0.7396	0.7277	0.4289	0.3705
	ARCACD	0.7884	0.8487	0.3823	0.2740	0.7682	0.7596	0.4007	0.3338
RCD	Base	0.7249	0.7618	0.4271	0.3518	0.7595	0.7589	0.4052	0.3327
	CACD	0.7284	0.7584	0.4291	0.3670	0.7706	0.7631	0.3986	0.3174
	ARCACD	0.7787	0.8384	0.3920	0.2747	0.7715	0.7634	0.3971	0.3168
SCD	Base	0.7257	0.7621	0.4263	0.3594	0.7570	0.7499	0.4097	0.3572
	CACD	0.7273	0.7599	0.4320	0.3358	0.7702	0.7570	0.4045	0.3506
	ARCACD	0.7782	0.8387	0.3956	0.2632	0.7704	0.7583	0.4043	0.3501

residuals with the CACD contrastive loss, DINA, IRT, MIRT, RCD, and SCD obtain AUC values of approximately 0.838–0.843. On the Junyi subset, ARCAD also yields consistent improvements on all four response metrics across six backbones. Overall, in the absence of real affect labels, joint modeling of ability residuals and contrastive affective representations yields consistent response-prediction gains across the reported datasets and backbones.

6.2 Affect Alignment Analysis (RQ2)

This section answers RQ2, namely whether explicit ability residuals can help the affective branch better align with real affect when real affect labels are available. Table 7 reports affect metrics and presents ASSIST2017 and ASSIST2012 side by side. Since Base does not contain an affective branch, the table includes only ACD and ARACD trained with real affect label supervision. The metrics are Affect RMSE and Affect MAE.

Table 7 Affect alignment results under real affect labels.

Backbone	Method	ASSIST2017		ASSIST2012	
		Affect RMSE ↓	Affect MAE ↓	Affect RMSE ↓	Affect MAE ↓
DINA	ACD	0.2614	0.1914	0.2184	0.1502
	ARACD	0.1990	0.1489	0.1937	0.1239
IRT	ACD	0.2461	0.1785	0.2453	0.1792
	ARACD	0.1925	0.1389	0.1919	0.1179
MIRT	ACD	0.2640	0.1900	0.2495	0.1814
	ARACD	0.1922	0.1390	0.1913	0.1203
NCDM	ACD	0.2266	0.1640	0.2422	0.1703
	ARACD	0.1944	0.1409	0.1922	0.1196
RCD	ACD	0.2055	0.1441	0.1966	0.1245
	ARACD	0.2038	0.1571	0.1942	0.1213
SCD	ACD	0.2054	0.1436	0.1996	0.1269
	ARACD	0.2037	0.1532	0.1945	0.1246

Table 7 provides generally supportive but non-uniform evidence for improved affect alignment after introducing ability residuals. On ASSIST2012, ARACD obtains lower Affect RMSE and Affect MAE than the corresponding ACD on all six backbones. On ASSIST2017, ARACD lowers both affect metrics for DINA, IRT, MIRT, and NCDM, and reduces Affect RMSE for RCD and SCD, although Affect MAE increases on these two backbones. These findings highlight that ability residuals reduce the pressure on the affective branch to absorb cognitive bias in many settings. Still, affect-alignment gains should not be interpreted as guaranteed for every backbone and metric. This result also indicates that similar response metrics do not necessarily imply affective representation alignment; therefore, later contamination diagnostics separately analyze whether affective representations absorb cognitive residuals.

6.3 Ablation Study (RQ3)

This section answers RQ3 by comparing the full model with three variants: Only stu-item residual, Only concept residual, and w/o ability residual. ASSIST2012/ASSIST2017 use supervised ARACD training, whereas ASSIST2009/Junyi use ARCACD training without affect labels. All variants are evaluated under the same data split and model-selection protocol within each dataset.

Table 8 Hierarchical ablation of supervised ability residual layers on ASSIST2017 and ASSIST2012.

Variant	ASSIST2017						ASSIST2012					
	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$	Affect RMSE $\downarrow$	Affect MAE $\downarrow$	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$	Affect RMSE $\downarrow$	Affect MAE $\downarrow$
Full ARACD	0.7394	0.8078	0.4203	0.3461	0.1944	0.1409	0.7487	0.7617	0.4138	0.3316	0.1922	0.1196
Only stu-item residual	0.7385	0.8051	0.4219	0.3494	0.1988	0.1456	0.7444	0.7537	0.4170	0.3328	0.1942	0.1210
Only concept residual	0.7288	0.7994	0.4286	0.3839	0.2082	0.1484	0.7457	0.7526	0.4156	0.3321	0.1973	0.1258
w/o ability residual	0.7230	0.7938	0.4296	0.4018	0.2266	0.1640	0.7421	0.7502	0.4185	0.3336	0.2422	0.1703

Table 8 reveals that in the supervised setting, the complete model achieves the best result on every reported response and affect metric across ASSIST2017 and ASSIST2012. The student-item-only and concept-only variants are consistently weaker than the complete model, indicating that the two residual layers provide complementary information.

Table 9 Hierarchical ablation of ability residual layers on ASSIST2009 and Junyi.

Variant	ASSIST2009				Junyi			
	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$	ACC $\uparrow$	AUC $\uparrow$	RMSE $\downarrow$	MAE $\downarrow$
Full ARCACD	0.7884	0.8487	0.3823	0.2740	0.7682	0.7596	0.4007	0.3232
Only stu-item residual	0.7804	0.8388	0.3872	0.2909	0.7599	0.7517	0.4045	0.3354
Only concept residual	0.7333	0.7872	0.4116	0.3388	0.7636	0.7478	0.4032	0.3338
w/o ability residual	0.7253	0.7573	0.4332	0.3845	0.7396	0.7277	0.4289	0.3705

Accordingly, Table 9 reveals a similar pattern in the setting without affect labels. On both ASSIST2009 and Junyi, the full model achieves the best ACC/AUC and the lowest RMSE/MAE among all variants. The student-item layer explains most of the gain, while the concept layer provides complementary concept-related corrections when combined with it.

6.4 Hyperparameter Analysis (RQ4)

This section answers RQ4 by examining two capacity parameters on NCDM: the attention dimension d_{att} and the student-item low-rank dimension $rank$. ASSIST2017 uses supervised affect prediction, while ASSIST2009 uses contrastive ARCACD without affect labels. Figures 3 and 4 report the complete sensitivity comparisons: Figure 3 uses fixed $rank = 1024$ and sweeps $d_{att} = 4, 8, 16, 32$, while Figure 4 uses fixed $d_{att} = 16$ and sweeps $rank$ from 4 to 2048.

Figure 3 demonstrates that d_{att} has a limited but consistent effect once $rank = 1024$ is fixed. Both datasets obtain their best or near-best response metrics around

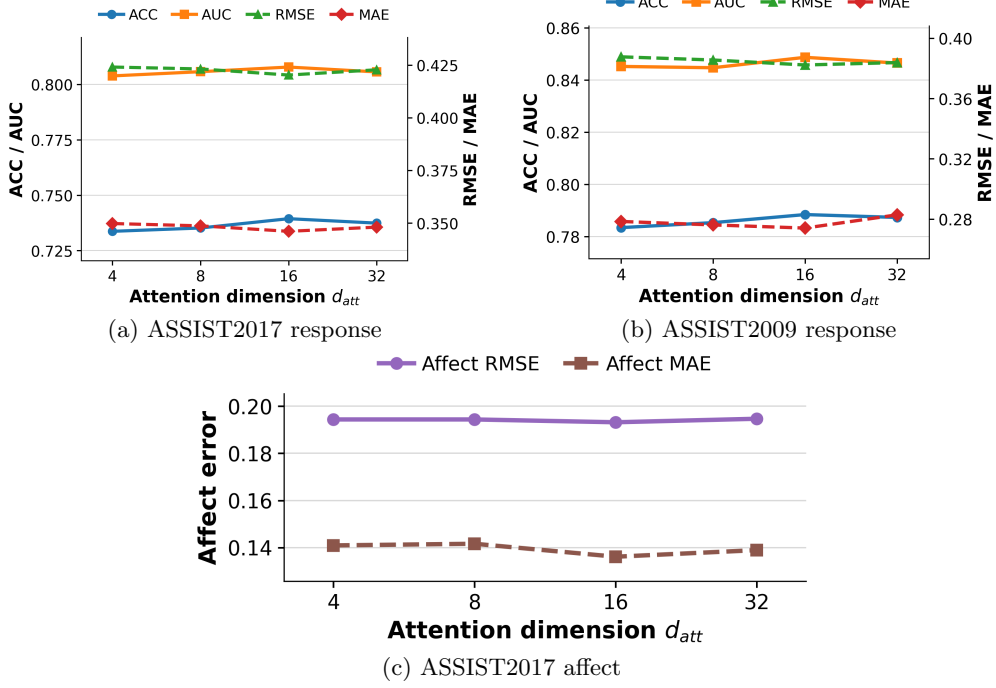


Fig. 3 Sensitivity to the concept-attention dimension d_{att} with fixed $rank = 1024$. Panels (a) and (b) report ACC/AUC and RMSE/MAE; panel (c) reports Affect RMSE/MAE.

$d_{att} = 16$, while increasing it to 32 introduces no stable gain. Thus, concept residual attention only needs moderate capacity when the student-item residual layer is sufficiently expressive.

Figure 4 highlights that $rank$ is the more important capacity source. Increasing $rank$ from 4 to 1024 markedly improves ASSIST2009 response metrics and mildly improves ASSIST2017, while $rank = 2048$ yields limited additional benefit and can increase some errors. Affect prediction on ASSIST2017 follows the same pattern, with lower error around $rank = 1024$. Overall, $rank = 1024$ provides a practical balance between response accuracy, affect alignment, and capacity saturation.

6.5 Residual Map Analysis (RQ5)

This section answers RQ5 by visualizing the residual space using $r_{si}^{stu-item}$ and $r_{si}^{concept}$ as two-dimensional coordinates. Figure 5 presents the sampled test interactions from all four datasets, colored by true response labels. The residual spaces exhibit response-related directions rather than random perturbations: ASSIST2017 and ASSIST2009 are dominated by the student-item residual direction, ASSIST2012 shows a smoother but still visible offset, and Junyi uses both residual types. This provides representation-level evidence that the two residual layers absorb complementary cognitive bias.

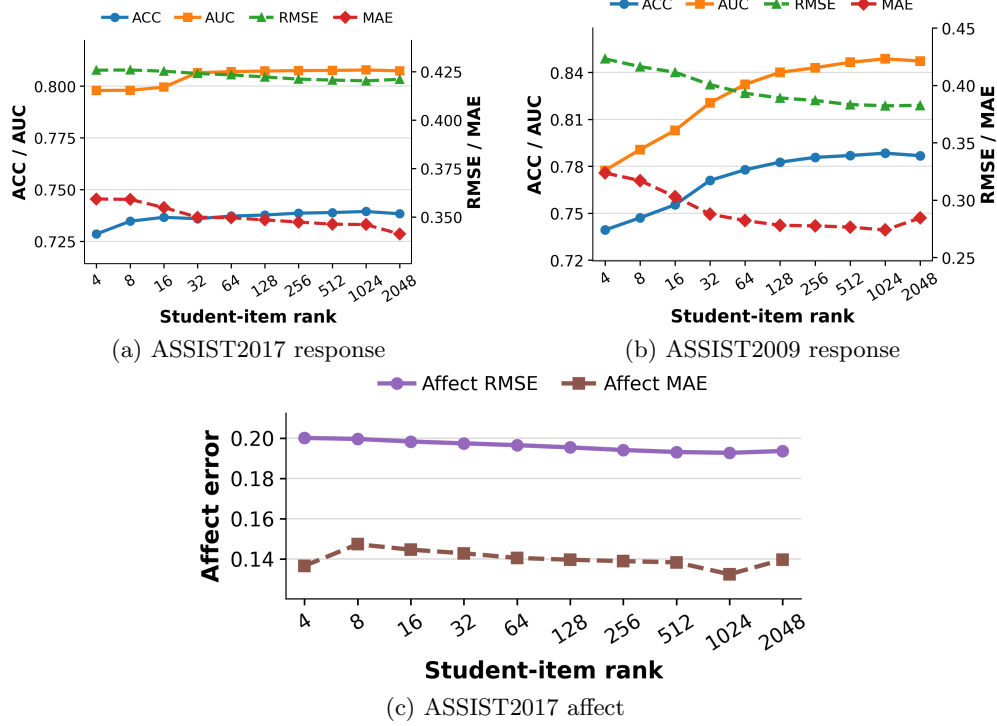


Fig. 4 Sensitivity to the student-item low-rank dimension $rank$ with fixed $d_{att} = 16$. Panels (a) and (b) report ACC/AUC and RMSE/MAE; panel (c) reports Affect RMSE/MAE.

6.6 Affect Contamination Diagnostics (RQ6)

This section answers RQ6 by testing whether ability residuals reduce cognitive leakage into affective representations. Since response metrics alone cannot determine whether the affective branch has learned a semantically clean non-cognitive state, we construct a synthetic contamination probe. A known bias term is added to the response-generation process, composed of student, item, and concept effects, and ARACD is trained with $d_{att} = 16$ and $rank = 1024$ without exposing the bias term. Then we test whether this term is easier to recover from the affective branch or the ability residual branch. Three diagnostics are reported: mean abs corr(*affect*, *bias*), affect probe R^2 , and mean abs corr(*ability residual*, *bias*).

Table 10
Affect contamination leakage diagnostics on ASSIST2017.

Variant	Affect RMSE ↓	Affect MAE ↓	Mean abs corr (<i>affect</i> , <i>bias</i>) ↓	Affect probe R^2 ↓	Mean abs corr (<i>residual</i> , <i>bias</i>) ↑
Full	0.1935	0.1383	0.0665	0.0568	0.6890
w/o A residual	0.2298	0.1664	0.5878	0.4716	n/a

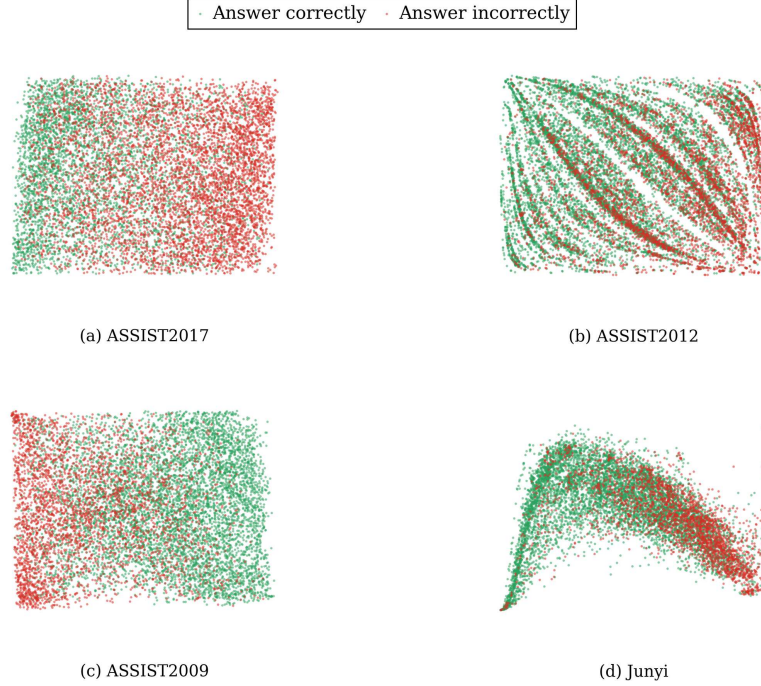


Fig. 5 Ability-residual interaction maps for ASSIST2017, ASSIST2012, ASSIST2009, and Junyi.

Table 10 and Figure 6 provide evidence consistent with a redistribution of synthetic cognitive bias away from the affective branch. Without ability residuals, the mean $\text{abs corr}(\text{affect}, \text{bias})$ and affect probe R^2 reach 0.5878 and 0.4716, respectively. With ability residuals, they drop to 0.0665 and 0.0568, while residual-bias correlation reaches 0.6890, and Affect RMSE/MAE also decrease. These diagnostics combined suggest that more of the synthetic contamination signal is represented in the ability-residual pathway rather than in the affective representation.

As a complementary view, we project supervised ARACD affect representations from ASSIST2017 into a PCA space and color them by response label and ability residual magnitude. Ideally, the affective representation should preserve response-relevant non-cognitive variation needed for guess/slip modulation, while not being primarily structured by the magnitude of ability residuals.

Figure 7 and Table 11 support this expectation, as the affect dimensions correlate only weakly with absolute ability residuals (at most 0.1167), while the first three dimensions show clearer response-label correlations. This does not imply a fully isolated affect space, since affect still participates in response prediction. Still, it indicates that ability residuals reduce the tendency of the affect branch to encode stable cognitive bias.

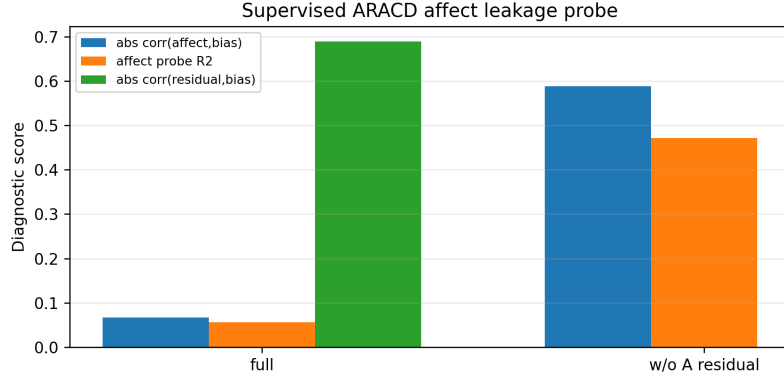


Fig. 6 Contamination leakage probe of the affective branch.

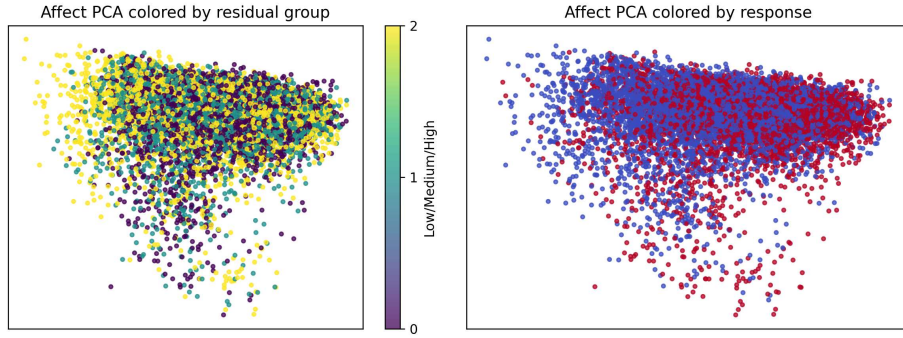


Fig. 7 PCA visualization of supervised ARACD affective representations on ASSIST2017, colored by ability-residual group (left) and response label (right).

Table 11 Absolute correlations between supervised ARACD affect dimensions and residuals/response labels.

Affect dim	Abs corr. with abs residual ↓	Abs corr. with response ↑
0	0.1167	0.2114
1	0.0958	0.2081
2	0.0785	0.1933
3	0.0052	0.0011

6.7 Long-Tail Student Results (RQ7)

This section answers RQ7 by evaluating low-activity students, defined by the lower tertile of training interactions. These students are challenging cases because sparse records provide limited evidence for estimating mastery states, affective representations, and personalized residuals. In this scenario, we compare ACD/RCD/SCD/ARACD on ASSIST2017 and CACD/RCD/SCD/ARCACD on ASSIST2009, using ACC, AUC, RMSE, ECE, and Brier. ECE measures the confidence-accuracy gap over probability bins (Pakdaman Naeini et al. 2015), while Brier score measures the mean squared error between predicted probability and the binary label (Brier 1950); lower values are better for both.

Table 12
Results for long-tail students.

Dataset	Method	Low-activity ACC $\uparrow$	Low-activity AUC $\uparrow$	Low-activity RMSE $\downarrow$	Low-activity ECE $\downarrow$	Low-activity Brier $\downarrow$
ASSIST2017	ACD	0.6915	0.7442	0.4547	0.0528	0.2067
ASSIST2017	RCD	0.6922	0.7478	0.4479	0.0263	0.2007
ASSIST2017	SCD	0.6887	0.7466	0.4481	0.0209	0.2008
ASSIST2017	ARACD	0.7001	0.7650	0.4405	0.0175	0.1941
ASSIST2009	CACD	0.6550	0.7365	0.4596	0.0772	0.2112
ASSIST2009	RCD	0.7072	0.7497	0.4404	0.0303	0.1939
ASSIST2009	SCD	0.7095	0.7489	0.4413	0.0394	0.1948
ASSIST2009	ARCACD	0.7097	0.7533	0.4398	0.0381	0.1934

Table 12 reveals that ARACD performs best on all reported low-activity metrics in ASSIST2017. On ASSIST2009, ARCACD achieves the highest AUC and the lowest RMSE/Brier, while ACC and ECE remain close to strong RCD/SCD baselines. These results indicate that ability residuals are especially helpful for ranking and probability-error reduction in sparse student groups.

6.8 Case Study

We further analyze student S453 from ASSIST2017 to illustrate how ability residuals correct individual response bias. This student has 388 training interactions and 109 test interactions. After retraining the independent Base NCDM used in the case analysis, Base NCDM obtains ACC/MAE/AUC of 0.6606/0.4371/0.7361 on this student. Notably, ACD still leaves several systematic probability errors for this student, with ACC/MAE/AUC of 0.6330/0.4729/0.7031, while ARACD raises ACC to 0.7523, MAE reduces to 0.3529, and AUC improves to 0.7571, correcting 16 ACD errors while introducing only 3 new errors. This suggests that ability residuals provide directional personalized calibration rather than uniform probability shifts.

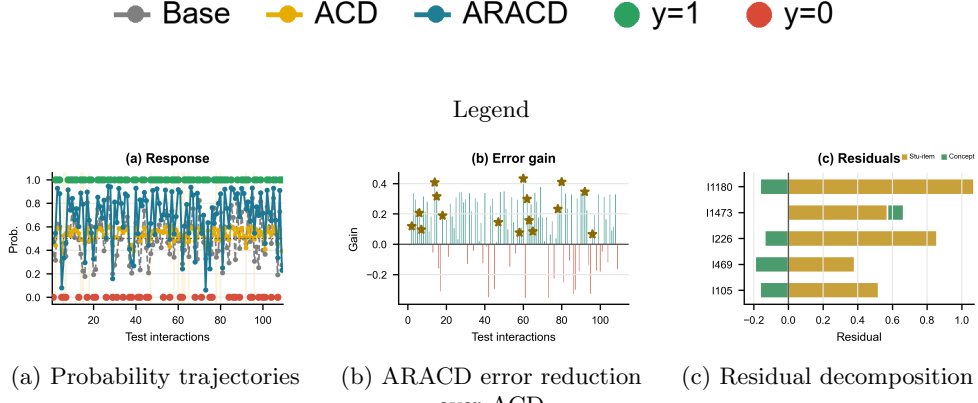


Fig. 8 Student-level case study for S453. Panels (a)–(c) show probability trajectories, ARACD error reduction over ACD, and residual decomposition.

Table 13
Representative interactions and residual decomposition for student S453.

Attempt	Item	Concepts	y	Base prob	ACD prob	ARACD prob	$r^{stu-item}$	$r^{concept}$
60	1180	38	1	0.4787	0.4996	0.9167	1.0712	-0.1631
80	1473	15	1	0.7224	0.4862	0.8808	0.5718	0.0934
14	226	35	1	0.3678	0.4945	0.8876	0.8574	-0.1367
62	135	46	0	0.2850	0.5248	0.2414	-0.3903	-0.1380
29	40	14	0	0.1993	0.4367	0.1582	-0.6087	-0.2034
6	1171	38	0	0.4621	0.5244	0.3337	-0.2446	-0.1631

Table 13 presents representative corrected interactions. For $y = 1$, positive student-item residuals raise ACD probabilities near 0.49 to 0.8808–0.9167; for $y = 0$, negative residuals lower probabilities to 0.1582–0.3337. The opposite residual directions across labels show that ARACD performs bidirectional student-item calibration.

Table 14
Affect alignment statistics for student S453.

Affect dim	Mean true	ACD pred	ARACD pred	ACD MAE ↓	ARACD MAE ↓
A1	0.2133	0.1293	0.2430	0.1107	0.0966
A2	0.6941	0.7486	0.6906	0.1138	0.1092
A3	0.0591	0.0701	0.0924	0.1083	0.1295
A4	0.0651	0.2101	0.1202	0.2251	0.1507

Figure 9 and Table 14 further demonstrate that ARACD preserves affect alignment for S453: overall affect MAE decreases from 0.1395 to 0.1215, with improvements on A1, A2, and A4. Thus, the response improvement does not come at the expense of affect prediction; ability residuals absorb stable student-item offsets while reducing

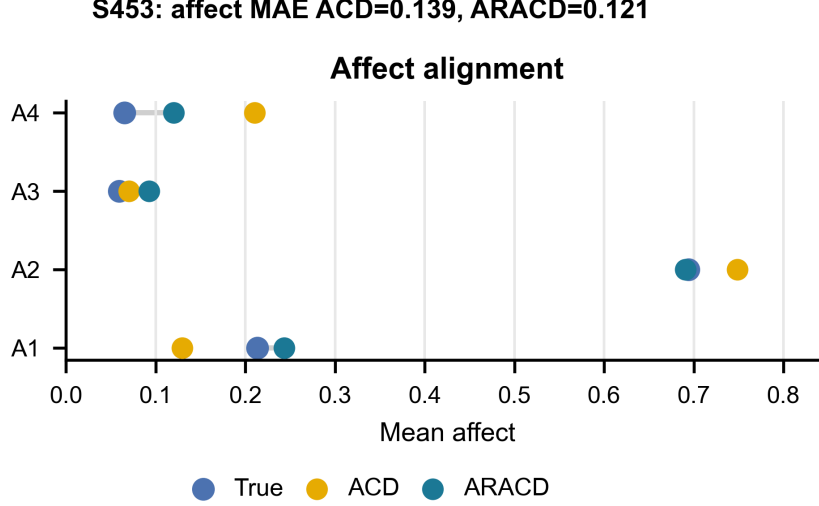


Fig. 9 Student-level affect analysis for S453. Mean values of the four observed affect dimensions are compared with ACD and ARACD predictions.

pressure on the affective representation. Overall, this case study shows that ability residuals can correct individualized response bias while keeping affect predictions aligned with external labels.

7 Conclusion

This paper studies the affect contamination problem in affective cognitive diagnosis. Unlike approaches that treat the affective module mainly as a compensator for backbone prediction errors, this study focuses on the base cognitive diagnosis backbone, which may still leave stable cognitive bias unmodeled. Indeed, if such bias is not explicitly modeled, affective representations may absorb information about items, concepts, student ability, or student-item matching, making them less specific to non-cognitive states. Motivated by this observation, this work introduces an ability-residual decoupling framework that assigns stable cognitive residuals first to the student-item residual layer and the concept residual layer, and then to the affective module model guess/slip modulation. This provides a structured separation between cognitive residuals and non-cognitive factors.

Experimental results support the proposed approach from multiple perspectives. Specifically, the main experiments highlight that the ability-residual framework can be combined with supervised ACD (with real affect labels) or contrastive CACD (without real affect labels), yielding response-prediction gains across all tested backbones in the reported main comparisons. On datasets with real affect labels, ARACD generally improves affect alignment, although not every backbone and metric benefits. Hierarchical ablation further suggests that the student-item residual layer is the main source of personalized response correction, while the concept residual layer complements

concept-related bias. Furthermore, the experiments reveal that Q-matrix-constrained attention can aggregate relevant concept residuals in an interpretable way. Contamination leakage diagnostics reveal that removing ability residuals substantially increases the correlation and linear decodability between affective representations and synthetic cognitive bias. In contrast, the complete model transfers more of this stable bias to the ability residual pathway. PCA visualization, long-tail student reanalysis, and student-level cases further indicate that the framework not only improves overall prediction performance but also gives a more explicit separation of modeling roles in representation organization, low-interaction student diagnosis, and student-level correction.

Overall, this paper concludes that affective cognitive diagnosis should not focus only on improving response prediction but should also examine whether affective representations contain cognitive residuals that do not belong to affective states. Ability-residual decoupling provides a structured method for this problem, allowing the model to explain stable cognitive residual variation and model non-cognitive modulation, thereby improving the reliability, interpretability, and response prediction stability of affective diagnosis. Future work may further incorporate real multimodal affect observations and cross-platform educational data to evaluate the generalizability and limits of ability-residual decoupling in more complex learning scenarios.

Statements and Declarations

Funding. No funding was received for conducting this study.

Competing interests. The authors have no relevant financial or non-financial interests to disclose.

Ethics approval and consent to participate. Not applicable. This study used publicly available, de-identified educational datasets and involved no new data collection from human participants.

Consent for publication. Not applicable.

Data availability. The public datasets analysed in this study (ASSIST2017, ASSIST2012, ASSIST2009, and Junyi) are available from their original public repositories. The source code is available at <https://github.com/psychosiwa/ARACD>. Additional processed data and experimental outputs are available from the corresponding author on reasonable request.

Materials availability. Not applicable.

Code availability. The source code is publicly available at <https://github.com/psychosiwa/ARACD>.

Author contributions. Boyuan Zhao: methodology, software, validation, formal analysis, investigation, data curation, visualization, and writing—original draft. Meng Ye: conceptualization, supervision, project administration, and writing—review and editing. Both authors read and approved the final manuscript.

References

- Aggarwal CC (2016) Recommender Systems: The Textbook. Springer, Cham, Switzerland
- Alain G, Bengio Y (2016) Understanding intermediate layers using linear classifier probes. arXiv:1610.01644
- Baker RSJd, D’Mello SK, Rodrigo MMT, et al (2010) Better to be frustrated than bored: The incidence, persistence, and impact of learners’ cognitive-affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies* 68(4):223–241
- Bengio Y, Courville A, Vincent P (2013) Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(8):1798–1828
- Brier GW (1950) Verification of forecasts expressed in terms of probability. *Monthly Weather Review* 78(1):1–3
- Calvo RA, D’Mello S (2010) Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing* 1(1):18–37
- Chen T, Kornblith S, Norouzi M, et al (2020) A simple framework for contrastive learning of visual representations. In: *Proceedings of the International Conference on Machine Learning*, vol 119. PMLR, pp 1597–1607
- Conati C, Maclaren H (2009) Empirically building and evaluating a probabilistic model of user affect. *User Modeling and User-Adapted Interaction* 19(3):267–303
- D’Mello S, Graesser A (2012) Dynamics of affective states during complex learning. *Learning and Instruction* 22(2):145–157
- D’Mello SK, Picard RW, Graesser AC (2007) Toward an affect-sensitive AutoTutor. *IEEE Intelligent Systems* 22(4):53–61
- Feng M, Heffernan N, Koedinger KR (2009) Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction* 19(3):243–266
- Gao W, Liu Q, Huang Z, et al (2021) RCD: Relation map driven cognitive diagnosis for intelligent education systems. In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp 501–510
- Geirhos R, Jacobsen JH, Michaelis C, et al (2020) Shortcut learning in deep neural networks. *Nature Machine Intelligence* 2:665–673

- He K, Zhang X, Ren S, et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 770–778
- Huang T, Chen Y, Geng J, et al (2025) A higher-order neural cognitive diagnosis model with hierarchical attention networks. *Expert Systems with Applications* 273:126848
- Junker BW, Sijtsma K (2001) Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement* 25(3):258–272
- Khosla P, Teterwak P, Wang C, et al (2020) Supervised contrastive learning. *Advances in Neural Information Processing Systems* 33:18661–18673
- Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. *Computer* 42(8):30–37
- Leighton JP, Gierl MJ (2007) *Cognitive Diagnostic Assessment for Education: Theory and Applications*. Cambridge University Press, Cambridge, UK
- Locatello F, Bauer S, Lucic M, et al (2019) Challenging common assumptions in the unsupervised learning of disentangled representations. In: Proceedings of the International Conference on Machine Learning, vol 97. PMLR, pp 4114–4124
- Lord FM (1980) *Applications of Item Response Theory to Practical Testing Problems*. Lawrence Erlbaum Associates, Hillsdale, NJ, USA
- Pakdaman Naeini M, Cooper G, Hauskrecht M (2015) Obtaining well calibrated probabilities using Bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence* 29(1)
- Pekrun R (2006) The control-value theory of achievement emotions: Assumptions, corollaries, and implications for educational research and practice. *Educational Psychology Review* 18(4):315–341
- Piech C, Bassen J, Huang J, et al (2015) Deep knowledge tracing. *Advances in Neural Information Processing Systems* 28
- Reckase MD (2009) *Multidimensional Item Response Theory*. Springer, New York, NY, USA
- Rendle S (2010) Factorization machines. In: Proceedings of the IEEE International Conference on Data Mining, pp 995–1000
- Su Y, Shen S, Zhu L, et al (2024) Global and local neural cognitive modeling for student performance prediction. *Expert Systems with Applications* 237:121637

- Thai-Nghe N, Drumond L, Horvath T, et al (2010) Recommender system for predicting student performance. *Procedia Computer Science* 1(2):2811–2819
- de la Torre J (2009) DINA model and parameter estimation: A didactic. *Journal of Educational and Behavioral Statistics* 34(1):115–130
- VanLehn K (2011) The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist* 46(4):197–221
- Vaswani A, Shazeer N, Parmar N, et al (2017) Attention is all you need. *Advances in Neural Information Processing Systems* 30
- Vie JJ, Kashima H (2019) Knowledge tracing machines: Factorization machines for knowledge tracing. *Proceedings of the AAAI Conference on Artificial Intelligence* 33(1):750–757
- Wang F, Liu Q, Chen E, et al (2020) Neural cognitive diagnosis for intelligent education systems. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, pp 6153–6161
- Wang S, Zeng Z, Yang X, et al (2023) Self-supervised graph learning for long-tailed cognitive diagnosis. *Proceedings of the AAAI Conference on Artificial Intelligence* 37(1):110–118
- Wang S, Zeng Z, Yang X, et al (2024) Boosting neural cognitive diagnosis with student’s affective state modeling. *Proceedings of the AAAI Conference on Artificial Intelligence* 38(1):620–627
- Woolf BP, Burleson W, Arroyo I, et al (2009) Affect-aware tutors: recognising and responding to student affect. *International Journal of Learning Technology* 4(3/4):129–164