Knowing When to Defer: Selective Prediction for Responsible Knowledge Tracing

Joshua Mitton[†] Eedi
Prarthana Bhattacharyya[†] Eedi
Ralph Abboud Renaissance Philanthropy
Simon Woodhead Eedi

Abstract

Research on Knowledge Tracing (KT) models traditionally focuses on improving predictive accuracy. However, responsible real-world deployment requires models to know when to defer uncertain predictions to a human teacher. We introduce an intrinsic selective prediction layer for existing KT models using Monte Carlo Dropout (MC-Dropout) to quantify uncertainty. We evaluate this approach across three architectures (DKT, SAKT, and AKT) using the Eedi mathematics dataset. Abstaining on the 20% most uncertain predictions lifts accuracy by 2.3 to 3.0 percentage points, AUC by 1.9 to 2.4 percentage points and F1 by 1.4 to 4.3 percentage points without any retraining. This abstention strategy is highly targeted: the deferred set exhibits 1.45 to 1.60 times the error rate of the kept set. Furthermore, this targeting holds within every question-difficulty quartile and remains fair across student-ability levels. Importantly, MC-Dropout variance gives roughly five times the AUC lift of a calibrated two-parameter logistic (2PL) Item Response Theory (IRT) baseline as a selective-prediction signal. A variance decomposition of the model's epistemic uncertainty (BALD) reveals that the entire classical psychometric stack, comprising question difficulty, student ability, IRT-style outcome ambiguity, and historical curriculum coverage, explains less than 4% of the signal under linear modeling and at most 23% even with a non-linear regressor. This leaves 77% to 90% as architecture-specific epistemic content that MC-Dropout surfaces and simpler proxies cannot recover. Selective prediction with model-native epistemic uncertainty is therefore a necessary component of responsible KT deployment, complementary to subgroup-fairness audits and downstream classroom evaluation rather than a substitute for them.

Keywords: Knowledge tracing, Uncertainty, MC dropout.

1. Introduction

Knowledge Tracing (KT) models are widely used to predict how a student will respond to future questions based on their past interactions. These models have evolved significantly over time, transitioning from early Bayesian approaches Pardos and Heffernan (2010); Yudelson et al. (2013) to deep learning architectures built on LSTMs Piech et al. (2015), self-attention Pandey and Karypis (2019); Ghosh et al. (2020), and graphs Yang et al. (2020), and more recently to frameworks leveraging Large Language Models Wang et al. (2025). Despite these architectural leaps, actual performance gains in accuracy are often marginal Ozyurt et al. (2024). More importantly, existing KT models typically struggle with low specificity, which allows incorrect student answers to slip by undetected during real-world deployment Yamkovenko et al. (2025).

[†] Equal contribution.

We address this critical gap by treating responsible KT deployment as a selective prediction problem. To achieve this, we introduce an intrinsic selective prediction layer designed to integrate directly with existing KT models. The proposed system calculates its own uncertainty, abstains from making the most unreliable predictions, and gracefully defers those specific cases to a human teacher. We base our analysis on data collected from a live educational setting through the Eedi mathematics dataset. By applying Monte Carlo Dropout Gal and Ghahramani (2016) to quantify uncertainty across three distinct KT architectures (DKT, SAKT, and AKT), we offer four primary contributions:

- We demonstrate that selective prediction improves accuracy, AUC, and F1 scores across all three architectures without requiring any retraining. This abstention is highly targeted, with deferred predictions showing an error rate 1.45 to 1.60 times higher than the kept predictions. The targeting holds true within every quartile of question difficulty and remains equitable across different student ability levels.
- We show that MC-Dropout variance outperforms a calibrated two-parameter logistic (2PL) Item Response Theory (IRT) baseline on the same selective prediction task, giving roughly five times the AUC lift at 80 percent coverage.
- We use variance decomposition of the model's epistemic uncertainty (BALD) to show that the entire classical psychometric stack (question difficulty, student ability, IRT-style outcome ambiguity, and historical curriculum coverage at four granularities) accounts for less than 4 percent of the signal under linear modeling and at most 23 percent even with a non-linear regressor (5-fold cross-validation, random forest). The remaining 77 to 90 percent is architecture-specific epistemic content: the model's uncertainty over its own learned representations of a student's unique sequential trajectory, which simpler models cannot capture even with the freedom of non-linear modeling.
- Finally, we establish that responsible KT deployment must rely on the model's intrinsic uncertainty signal, rather than falling back on simple psychometric proxies or heuristic coverage rules.

2. Related Works

Deep Knowledge Tracing Models Knowledge Tracing (KT) models the temporal dynamics of student learning by predicting knowledge states based on sequences of interactions with educational content Corbett and Anderson (2005). Various machine learning methods, including deep neural networks, have been proposed to address this sequential prediction task. Among these, deep sequential models adopt auto-regressive architectures to capture the progression of student interactions. Deep Knowledge Tracing (DKT) Piech et al. (2015) was one of the first to use an LSTM to estimate students' knowledge states over time. Subsequent work, such as KQN Lee and Yeung (2019), enhanced KT by encoding student learning activities into knowledge state and skill vectors and modeling their interactions with the dot product. Other lines of work introduce memory-augmented (DKVMN Zhang et al. (2017)) and graph-based representations of knowledge components (e.g., GIKT Yang et al. (2021); Liu et al. (2020)).

Attention-Based Architectures However, recurrent and attention-based models remain the dominant deployed families and are the focus of this study. Attention-based models leverage attention mechanisms to dynamically weigh past interactions. SAKT Pandey and Karypis (2019) introduced a self-attention architecture to model the relevance of historical responses. AKT Ghosh et al. (2020) introduced monotonic attention modules to model forgetting behavior.

Uncertainty Modeling in KT Despite these advances, uncertainty modeling in KT remains underexplored. Most existing models produce point predictions without quantifying the confidence in knowledge estimates. This is an important limitation for real-world educational applications. UKT Cheng et al. (2025) is the only other work to the best of our knowledge that introduces a KT model to capture uncertainty in student learning by representing knowledge states as stochastic distributions and modeling their transitions with a Wasserstein self-attention mechanism. While UKT provides an approach to modeling uncertainty, it introduces an entirely new architecture, which limits its applicability in real-world settings where existing KT models are already deployed. Adding uncertainty quantification on top of such models remains non-trivial with this method.

Learning to Defer Outside the KT literature, the broader machine-learning community has formalized the abstention-and-defer paradigm we operationalize here under the umbrella of learning to defer (Madras et al., 2018; Mozannar and Sontag, 2020). In this paradigm, a classifier explicitly chooses between predicting and routing to a human expert. Our contribution adapts this approach to a deployed KT setting using model-native epistemic uncertainty as the routing signal rather than a learned deferral head.

3. Predictive Uncertainty for Knowledge Tracing Models

3.1. Knowledge Tracing Models

We adopt the standard formulation of knowledge tracing Piech et al. (2015), which predicts a student's performance on a future exercise based on their sequence of past interactions. Formally, we model a student's interaction history up to time t as a sequence of exercises, $\{e_i\}_{i=1}^{t}$. Each exercise is represented as a three-tuple, $e_i = (q_i, c_i, r_i)$. Here, q_i denotes the unique question identifier, c_i encapsulates the pedagogical features associated with q_i (such as knowledge component identifiers), and $r_i \in \{\text{Correct}, \text{Incorrect}\}$ indicates the binary correctness of the student's response.

Given this historical sequence and the context of an upcoming question, the KT model F_θ predicts the probability of a correct outcome $\hat{r}_{t+1}$ for the next exercise. Formally, this prediction is defined as:

$$\hat{r}_{t+1} = F_\theta\left(q_{t+1}, c_{t+1}, \{e_i\}_{i=1}^{t}\right)$$

3.1.1. Baseline KT Models

To benchmark KT approaches for mathematics education, we consider three models as instantiations of F_θ: Deep Knowledge Tracing (DKT) Piech et al. (2015), Self-Attentive Knowledge Tracing (SAKT) Pandey and Karypis (2019), and Attentive Knowledge Tracing (AKT) Ghosh et al. (2020). DKT models student learning with recurrent neural networks

(RNNs). It takes as input the sequence of question identifiers and responses (q_i, r_i) and, conditioned on the next question q_{t+1}, predicts the probability of a correct response $\hat{r}_{t+1}$ by capturing temporal dependencies in the interaction history. SAKT replaces recurrence with a self-attention mechanism, which allows the model to selectively attend to relevant past interactions when predicting performance on the next question. AKT extends this approach by introducing contextual attention that considers both the similarity between questions and an exponential decay to model student forgetting. Together these three architectures span the dominant families of modern KT models (recurrent and attention-based), letting us test whether our claims about uncertainty are architecture-invariant. These models are typically evaluated using point-estimate metrics such as accuracy, F1 or AUC, which quantify predictive performance but provide no measure of model uncertainty. This gap motivates the Bayesian formulation introduced next.

3.2. MCDropout for KT Uncertainty

We estimate predictive uncertainty using Monte Carlo Dropout (MC Dropout) Gal and Ghahramani (2016). At inference, dropout remains active and predictions are averaged over M stochastic forward passes. Following the information-theoretic approach based on Shannon entropy Cover and Thomas (2006), we quantify total uncertainty by the entropy of the aggregated predictive distribution:

$$H\big(p(r_{t+1} \mid q_{t+1}, \mathcal{H}_t)\big) = -\sum_k \bar{p}_k \log \bar{p}_k, \quad \bar{p}_k = \frac{1}{M}\sum_{m=1}^{M} p_k^{(m)}.$$

where $\bar{p}_k$ denotes the mean predictive probability of class k estimated via MCDropout. Here, $\mathcal{H}_t = \{e_i\}_{i=1}^{t}$ represents the interaction history of a student up to time t, with each exercise $e_i = (q_i, c_i, r_i)$ defined by the question identifier q_i, its associated features or knowledge components c_i, and the observed response r_i. Higher entropy corresponds to greater uncertainty in the model's prediction, whereas lower entropy reflects more confident predictions. We additionally use the variance and standard deviation of the predictions across MC samples as complementary uncertainty measures to operationalize the selective prediction layer (Appendix A.4).

4. Experiments and Results

4.1. Architectures and dataset

We implemented our selective prediction approach across three KT architectures: DKT Piech et al. (2015), SAKT Pandey and Karypis (2019), and AKT Ghosh et al. (2020). These models represent the dominant recurrent and attention-based families in current KT literature.

The training dataset consists of student responses to mathematics questions. To standardize the context, we limited the interaction history to 100 question-answer pairs per student. Within this platform, students complete questions in five-question quiz sessions that progressively increase in difficulty. Consequently, the 100-interaction history typically reflects a student's performance across 20 distinct quizzes. Further dataset details are provided in Section A.1.

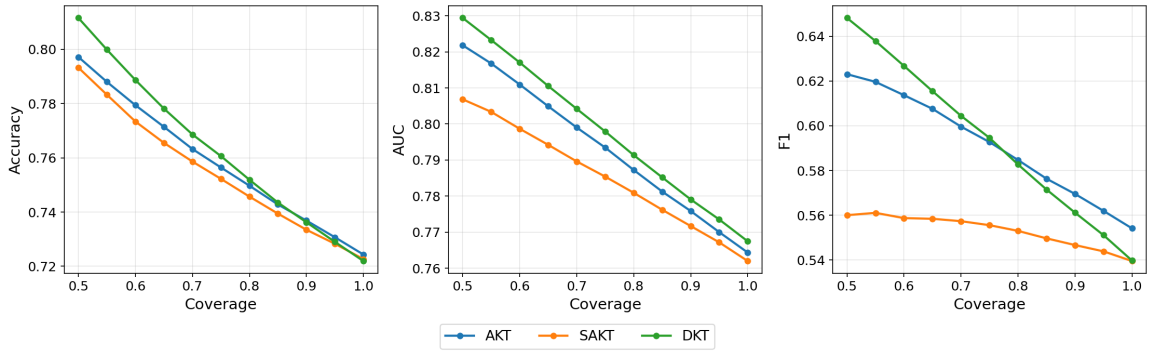

Figure 1: Selective prediction with MC-Dropout variance: every metric improves monotonically as we defer the most-uncertain cases, on all three architectures.

4.2. Qualitative uncertainty patterns

Across all three architectures, a clear qualitative pattern emerges. Misclassified predictions concentrate at higher MC-Dropout entropy and standard deviations than correctly classified ones. The uncertainty signal varies systematically with the within-quiz difficulty progression. Appendix C provides the distribution and trajectory plots that establish this behavior. While a monotonic correlation between uncertainty and model error is encouraging, the critical challenge is translating this signal into a practical deployment layer.

4.3. Selective Prediction: Translating Uncertainty into a Deployment Layer

To operationalize the uncertainty signal, we sort predictions by their MC-Dropout variance in ascending order and abstain from the most uncertain fraction to achieve a target coverage rate c. Figure 1 plots accuracy, AUC, and F1 scores against coverage for the binary correctness target. Across all three architectures, every metric improves monotonically as coverage decreases, achieving these gains entirely without retraining. At an operating point of $c = 0.80$ (where 20% of cases are deferred to a teacher), MC-Dropout variance increases accuracy by 2.3 to 3.0 percentage points (pp), AUC by 1.9 to 2.4 pp, and F1 by 1.4 to 4.3 pp. All lifts are statistically significant, as the 95% bootstrap confidence intervals (200 resamples) never cross zero. These intervals are exceptionally tight across all architectures, with margins of error of just ± 0.13 pp for the change in accuracy, ± 0.10 pp for AUC, and ± 0.20 pp for F1.

Targeting and fairness. This abstention mechanism is highly targeted. At 80% coverage, the error rate of the deferred set is 1.45 to 1.60 times higher than that of the kept set across the three models. To ensure this effect is not simply an artifact of question difficulty, we recalculated the error ratio within each question-difficulty quartile (Figure 2b). The targeting holds true within every stratum: every combination of model and question-difficulty quartile maintains an error ratio greater than 1.0, with the strongest targeting occurring on the hardest questions (2.3 to 2.5 times higher error rates across all three architectures). Equally important, the abstention is fair across student ability. Figure 2a shows that the

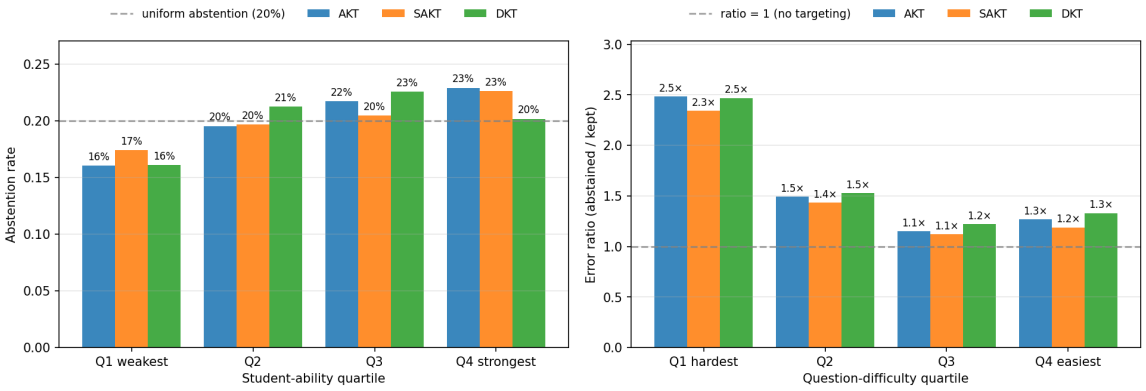

Figure 2: Selective prediction at $c = 0.80$ using the MC-Dropout variance signal is both fair and targeted. **(a)** Abstention rate by student-ability quartile: all four quartiles fall close to the 20% uniform baseline, and the weakest students are abstained on *less*, not more, than the rest. **(b)** Error ratio (abstained / kept) within each question-difficulty quartile: every (model × quartile) cell is above 1.0, with the strongest targeting on the hardest questions (2.3–2.5×).

abstention rate ranges tightly between 16% and 23% across all four student-ability quartiles, close to the 20% uniform baseline. The weakest students (Q1) actually receive the *lowest* abstention rate (16–17%), ruling out the failure mode in which a model defers low-ability students to teachers by default.

4.4. Comparison to a calibrated IRT baseline

A simpler alternative to our approach would be to abstain on predictions that a calibrated two-parameter logistic (2PL) Item Response Theory (IRT) (Lord et al., 1966) model identifies as ambiguous. The validation cohort is held out by student, so a pre-fit IRT model trained on the training split has no ability parameter for these previously-unseen learners. We therefore fit each validation student's ability θ_u from a held-out portion of their own response history. We use each user's first 30 questions as the fitting window, estimating θ_u via maximum likelihood with item parameters a_q, b_q taken from an IRT table pre-fitted on the training cohort over the question pool. The IRT abstention signal $1-2|p_{\text{IRT}}-0.5|$ is then evaluated on their remaining 70 targets, and the MC-Dropout comparison is restricted to the same matched subset (∼110,000 targets per model).[1] Figure 3 compares MC-Dropout variance, MC-Dropout entropy, and the IRT uncertainty baseline on this matched subset. MC-Dropout variance yields an AUC lift of 2.0 to 2.4 percentage points at 80% coverage. In contrast, the IRT baseline provides a lift of only 0.41 to 0.46 percentage points, roughly five times smaller. This demonstrates that MC-Dropout is not merely a complex re-implementation of IRT-style confidence. Instead, its stochastic forward passes uncover genuine epistemic information that a calibrated psychometric model fundamentally misses.

1. The headline lifts hold on this matched subset: MC-Dropout variance gives Δacc@80 = +2.4–3.0 pp, ΔAUC@80 = +2.0–2.4 pp, ΔF1@80 = +1.6–4.6 pp, within 0.3 pp of the full-sequence numbers.

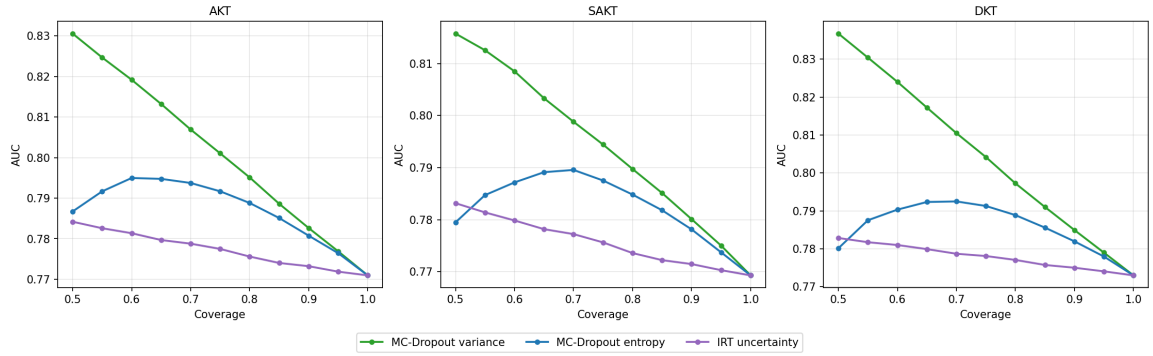

Figure 3: MC-Dropout variance dominates a calibrated 2PL IRT baseline as a selective-prediction signal across all three architectures (last-70 target subset).

We note that 2PL IRT is one specific baseline, bounded by its two-parameter representational capacity. Alternative selective-prediction baselines such as a temperature-scaled softmax of the deployed model itself (Guo et al., 2017) or a deep ensemble of independently trained checkpoints (Lakshminarayanan et al., 2017) would test the irreducibility claim more strongly, and remain a natural follow-up.

4.5. Drivers of predictive uncertainty

To determine what exactly this uncertainty tracks, we ask whether MC-Dropout's signal is recoverable from cheaper, model-free features of (student, question) pairs. We regress the model's *epistemic uncertainty* (BALD: total entropy minus expected aleatoric entropy across MC samples) on a sequence of nested predictor sets and report cumulative R^2 at each step. Predictors are added in order: question difficulty (empirical correctness rate), student ability (per-user accuracy), IRT-style outcome ambiguity ($H(p_{\text{IRT}})$), and curriculum coverage at four granularities (binary indicators for whether the target's subject, topic, subtopic, or construct appears in the student's first-30 context). The full procedure is detailed in Appendix E.

Across all three architectures, every classical psychometric factor yields negligible explanatory power. Question difficulty alone explains 0.4% to 1.5% of the variance. Adding student ability brings the cumulative total to 0.8% to 1.8%. IRT outcome ambiguity raises it to between 1.1% and 3.7%. Adding all four levels of curriculum coverage contributes a further 0.0% to 0.2%, leaving the cumulative linear-explained variance at most 3.8% across architectures (Figure 12, Appendix; see Appendix A.2 for overlap rates). To bound how much of this gap can be closed by a stronger model class, we re-fit the same predictor stack with a non-linear regressor (5-fold cross-validated random forest). Explained variance rises to 9.8% (SAKT), 10.5% (AKT), and 23.2% (DKT), still leaving **76.8% to 90.2%** unexplained.

This residual, dominant even under non-linear modeling, is exactly what MC-Dropout captures: the architecture's native uncertainty over its own learned, non-linear representations of a student's unique sequential trajectory. Heuristic coverage-tracking rules and

2PL-confidence calibrators are structurally incapable of recovering it. By construction they only have access to a small set of psychometric features (ability, difficulty, outcome ambiguity). Even given the freedom of a non-linear classifier, those features leave the majority of the signal architecture-specific. A model-native probe such as MC-Dropout is therefore necessary to surface this hidden uncertainty and enable responsible deployment.

5. Conclusion

Selective prediction with model-native epistemic uncertainty is a necessary component of responsible KT deployment. It is complementary to subgroup fairness audits, calibration analysis, and downstream classroom evaluation rather than a replacement for them. For confidence-aware systems to operate safely in real classrooms, they must reliably distinguish between safe and risky predictions using signals derived directly from their own internal representations. We have demonstrated that cheaper heuristic alternatives such as population statistics, IRT baselines, or curriculum coverage miss the vast majority of this per-prediction reliability signal by construction.

The model's native MC-Dropout epistemic uncertainty successfully targets and defers high-risk predictions across DKT, SAKT, and AKT architectures. The deferred set is also fair across student-ability and question difficulty quartiles. As a concrete deployment example, an Eedi-style next-question-recommendation pipeline could use the variance signal as a gate. Low-variance predictions would feed directly into the recommender's ranking. High-variance predictions would trigger a fallback (such as a teacher review queue, an easier diagnostic question, or a mastery quiz on the relevant construct), rather than committing to a low-confidence next-question pick. The architectural and analytical work in this paper grounds that integration. The operational work of validating it against student-outcome metrics in a real classroom remains future work.

Limitations. Our claims are bounded by four constraints. First, the headline lifts (2.3 to 3.0 pp accuracy at $c = 0.80$) are improvements on validation-set predictive metrics, not on student learning outcomes or teacher cognitive load. We have not run a classroom A/B test or simulated deployment, so the responsibility argument is conditional on those follow-ups. Second, our analysis uses a single dataset (Eedi mathematics responses), and the architecture-specific epistemic dominance has not been validated on other KT datasets. Third, the IRT comparison uses a 2-parameter logistic model fit on 30 questions per validation student. Alternative selective-prediction baselines such as a temperature-scaled softmax of the deployed model or a deep ensemble are not evaluated and remain promising future comparisons. Fourth, for the IRT baseline, we do not address cold-start: students with fewer than 30 prior responses fall outside our θ-fitting window.

References

Weihua Cheng, Hanwen Du, Chunxiao Li, Ersheng Ni, Liangdi Tan, Tianqi Xu, and Yongxin Ni. Uncertainty-aware knowledge tracing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i27.35007. URL https://doi.org/10.1609/aaai.v39i27.35007.

Albert T. Corbett and John R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4:253–278, 2005. URL https://api.semanticscholar.org/CorpusID:19228797.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006. ISBN 0471241954.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

Aritra Ghosh, Neil Heffernan, and Andrew S. Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 2330–2339, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3403282. URL https://doi.org/10.1145/3394486.3403282.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*, pages 1321–1330, 2017.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Jinseok Lee and Dit-Yan Yeung. Knowledge query network for knowledge tracing: How knowledge interacts with skills. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, LAK19, page 491–500, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362566. doi: 10.1145/3303772.3303786. URL https://doi.org/10.1145/3303772.3303786.

Yunfei Liu, Yang Yang, Xianyu Chen, Jian Shen, Haifeng Zhang, and Yong Yu. Improving knowledge tracing via pre-training question embeddings. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1577–1583. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/219. URL https://doi.org/10.24963/ijcai.2020/219. Main track.

Frederic M. Lord, Melvin R. Novick, and Allan Birnbaum. Some latent train models and their use in inferring an examinee's ability. 1966. URL https://api.semanticscholar.org/CorpusID:61082238.

David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning (ICML)*, pages 7076–7087, 2020.

Yilmazcan Ozyurt, Stefan Feuerriegel, and Mrinmaya Sachan. Automated knowledge concept annotation and question representation learning for knowledge tracing. *arXiv preprint arXiv:2410.01727*, 2024.

Shalini Pandey and George Karypis. A self-attentive model for knowledge tracing. In *EDM 2019 - Proceedings of the 12th International Conference on Educational Data Mining*, pages 384–389. International Educational Data Mining Society, 2019.

Zachary A Pardos and Neil T Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *International conference on user modeling, adaptation, and personalization*, pages 255–266. Springer, 2010.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 505–513, Cambridge, MA, USA, 2015. MIT Press.

Matias Valdenegro-Toro and Daniel Saromo Mori. A deeper look into aleatoric and epistemic uncertainty disentanglement. *arXiv preprint arXiv:2204.09308*, 2022.

Ziwei Wang, Jie Zhou, Qin Chen, Min Zhang, Bo Jiang, Aimin Zhou, Qinchun Bai, and Liang He. Llm-kt: Aligning large language models with knowledge tracing using a plug-and-play instruction. *arXiv preprint arXiv:2502.02945*, 2025.

Lisa Wimmer, Yusuf Sale, Paul Hofman, Bernd Bischl, and Eyke Hüllermeier. Quantifying aleatoric and epistemic uncertainty in machine learning: Are conditional entropy and mutual information appropriate measures? In *Uncertainty in Artificial Intelligence (UAI)*, pages 2282–2292, 2023.

Bogdan Yamkovenko, Charles Hogg, Maya Miller-Vedam, Phillip Grimaldi, and Walt Wells. Practical evaluation of deep knowledge tracing models for use in learning platforms. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 673–679, 2025.

Yang Yang, Jian Shen, Yanru Qu, Yunfei Liu, Kerong Wang, Yaoming Zhu, Weinan Zhang, and Yong Yu. Gikt: a graph-based interaction model for knowledge tracing. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 299–315. Springer, 2020.

Yang Yang, Jian Shen, Yanru Qu, Yunfei Liu, Kerong Wang, Yaoming Zhu, Weinan Zhang, and Yong Yu. Gikt: A graph-based interaction model for knowledge tracing. In Frank

Hutter, Kristian Kersting, Jefrey Lijffijt, and Isabel Valera, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 299–315, Cham, 2021. Springer International Publishing.

Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pages 171–180. Springer, 2013.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. Dkvmn. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 765–774, Republic and Canton of Geneva, CHE, 2017. International World Wide Web Conferences Steering Committee. ISBN 9781450349130. doi: 10.1145/3038912.3052580. URL https://doi.org/10.1145/3038912.3052580.

Appendix A. Experimental Setup

A.1. Dataset Details

We use a filtered subset of question-response logs from Eedi, an online mathematics learning platform. An interaction is a single question–response pair.
The corpus contains 4,257 unique questions. Each question is multi-choice, and has question-ID and question-text associated with it. For each question, there are four options to choose from: {A, B, C, D}. We have the labels for which of those options is correct, and the other three options are carefully curated distractors tied to a mathematics misconception. The additional features a question contains includes construct-text, construct-ID, explanation-text and misconception-text. A construct-text is the most granular level of knowledge related to question, for example, 'Construct 1: Find 10 less than a given number'. A misconception-text describes a cognitive error per distractor option which leads to mistakes. It is generic, unlike an explanation-text per option, which tends to be specific to the question.
We first filter students to those with at least 100 recorded responses and then retain the final 100 responses per student to standardize sequence length. The resulting training split contains 11,994 students (average 100 responses; 1,199,266 total interactions), and the validation split contains 1,493 students (average 100 responses; 149,278 total interactions) as shown in Table 1. All splits are student-disjoint: validation students do not appear in training. During evaluation, models predict each of the 100 student responses in sequence with access only to prior interactions (causal/left-to-right), never the current or future response.

Table 1: Dataset statistics after filtering. Interactions are question–response pairs. Splits are disjoint by student.

Split	# Students	Avg. responses	Total interactions
Train	11,994	100	1,199,266
Val	1,493	100	149,278

A.2. Curriculum Hierarchy and Context-Target Overlap

Each Eedi question is tagged with four nested levels of curriculum metadata. Ranging from coarsest to finest, these levels are *subject*, *topic*, *subtopic*, and *construct*. The curriculum forms a strict hierarchy, meaning every question belongs to exactly one category at each of these four levels. For example, question 31769 in our dataset falls under the subject 'Number', the topic 'Proportion', and the subtopic 'Direct Proportion'. At the most granular level, its construct is defined as 'Use direct proportion to solve non-unitary missing amounts in problems (e.g. recipes)'.

Table 2 details the total cardinality of each curriculum level. It also reports the average context-target overlap rate. We define this overlap as the fraction of a student's final 70 prediction targets that share a specific curriculum tag with at least one question from their initial 30-question context.

As expected, this coverage drops sharply as granularity increases. Because the dataset contains only six broad subjects, the majority of target questions share a subject with the

historical context, while only about 6% of targets share a specific construct. The variance decomposition presented in Section 4.3 explicitly controls for coverage across *all four* of these levels simultaneously. Consequently, our conclusion that curriculum coverage fails to drive uncertainty remains robust, regardless of the specific granularity at which a deployment system might track student progress.

Table 2: Curriculum hierarchy: cardinality and average context-target overlap rate (across the three architectures, on the last-70 prediction subset given a first-30 context).

Level	# unique values	% targets where level appears in context
Subject	6	81.4%
Topic	50	29.1%
Subtopic	212	14.1%
Construct	1,463	6.2%

A.3. Model Training

We train three knowledge tracing models: DKT Piech et al. (2015), SAKT Pandey and Karypis (2019), and AKT Ghosh et al. (2020), under a unified setup. Model sizes are summarized in Table 3. AKT is the largest (3.3 M parameters), DKT is the most compact (1.2 M). Unless otherwise noted, all models share the global training configuration in Table 4: Adam optimizer, learning rate 3×10^{-4}, cross-entropy loss, batch size 64, and 100 epochs, with a linear warmup followed by a cosine scheduler. Model-specific hyperparameters are listed in Table 5. We fix embedding and hidden dimensions to 128 and use one layer across models, with dropout tuned per architecture (0.5 for SAKT, 0.2 for AKT and DKT). AKT uses the `kq_same` setting. All other details are held constant to enable a fair comparison across architectures.

Reproducibility. MC-Dropout inference uses $M = 100$ stochastic forward passes per prediction with dropout active at the rates listed above; aleatoric and epistemic components are computed via the BALD decomposition (Section E). All training and inference uses a single fixed random seed; multi-seed robustness is not reported in this version.

Table 3: Parameter counts and estimated memory footprint for each model. We report total trainable parameters (in millions) and the estimated model size (in MB).

Model	Total params (M)	Est. size (MB)
DKT Piech et al. (2015)	1.2	4.890
SAKT Pandey and Karypis (2019)	1.7	6.990
AKT Ghosh et al. (2020)	3.3	13.020

Table 4: Global training hyper-parameters used across all models.

Setting	Value
Learning rate (LR)	0.0003
Optimizer	Adam
Loss	Cross-Entropy
Batch size	64
Epochs	100
Scheduler	Linear (warmup), cosine (training)

Table 5: Model-specific hyper-parameters.

	DKT	SAKT	AKT
Dropout	0.2	0.5	0.2
Embedding dim	128	128	128
Hidden dim	128	128	128
kq_same	-	-	True
Num layers	1	1	1

A.4. Model Evaluation

We use standard deviation of the model's predictions over Monte-Carlo samples as a metric for predictive uncertainty. The standard deviation per question per student is calculated as:

$$\sigma(p) = \mathbb{E}_k\big[\sigma(p_k)\big], \quad \sigma(p_k) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\Big[\big(p_k^{(m)} - \mu\big)^2\Big]}$$

where k denotes the class, $p_k^{(m)}$ denotes the predicted probability of sample m, μ is the mean predicted probability, and $\sigma(p_k)$ is the standard deviation of the model's predictions over M Monte-Carlo samples.

Appendix B. Model Quantitative Predictive Accuracy

On the validation split, all three architectures attain similar baseline accuracy on the binary correctness prediction task (see Table 6): AKT (72.44%), SAKT (72.27%), and DKT (72.20%). F1 also clusters closely: AKT yields the strongest F1 (55.42%), followed by DKT (53.98%) and SAKT (53.96%). AUC is similarly tight, ranging from 76.20% (SAKT) to 76.75% (DKT). These results suggest that while overall predictive performance is comparable across architectures, the more informative deployment-relevant metric is the quality of each model's per-prediction uncertainty signal, which we evaluate via selective prediction in Section 4.3.

Table 6: Baseline binary correctness prediction on the validation split (no abstention).

Model	Accuracy	F1	AUC
DKT Piech et al. (2015)	72.20	53.98	**76.75**
SAKT Pandey and Karypis (2019)	72.27	53.96	76.20
AKT Ghosh et al. (2020)	**72.44**	**55.42**	76.43

Appendix C. Uncertainty Analysis

C.1. Uncertainty by Prediction Correctness

The mean total entropy is consistently larger when the model's prediction is incorrect (Figure 4). This confirms that our uncertainty measure successfully flags specific model errors: which is precisely the operational relationship that selective prediction (Section 4.3) exploits.

However, when predictions are split by the *student's* outcome rather than the *model's* correctness, this relationship inverts. As shown in Figure 5 and Figure 6, both median entropy and prediction standard deviation are slightly higher on student-correct cases than on student-incorrect cases. This is an artifact of binary correctness prediction under class imbalance. On hard questions where a student is likely to fail, the model can confidently predict failure, clustering low-entropy mass on the student-incorrect side. On easier questions where the student succeeds, the model commits to a moderately confident correct prediction, pushing the entropy closer to the binary maximum.

Therefore, splitting by student outcome is not the appropriate diagnostic for model reliability; the model-correctness split (Figure 4) is the true indicator. Finally, we note that a perfectly calibrated binary classifier would not show such a drastic inversion. The magnitude of this skew reflects the model's tendency to make highly confident failure predictions on difficult questions. This calibration property warrants further investigation, particularly before porting selective prediction to deployment settings where confident-failure predictions about specific students could trigger negative downstream consequences.

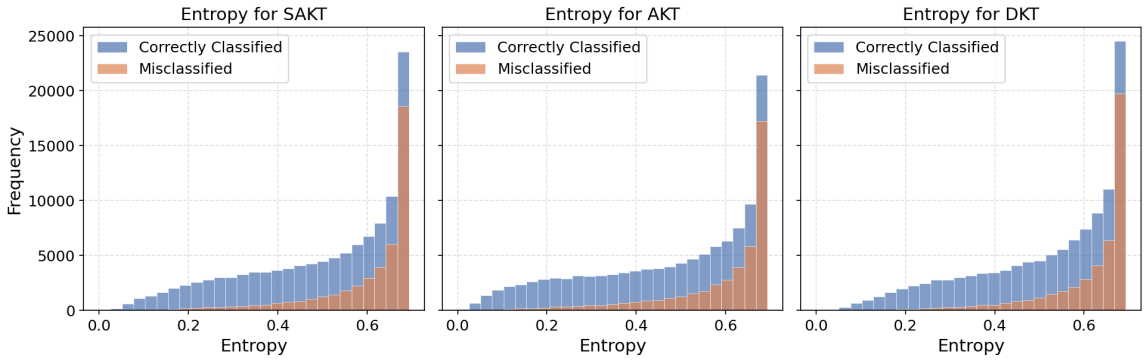

Figure 4: Total entropy distribution of each model's predictions as a function of their correctness. Misclassified data points clearly demonstrate a higher level of uncertainty.

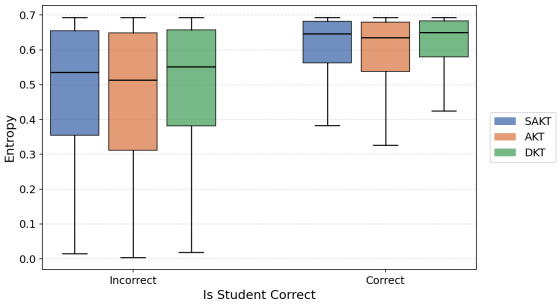

Figure 5: Box plot of total entropy of each model's predictions split into whether the student chose a correct response or an incorrect response.

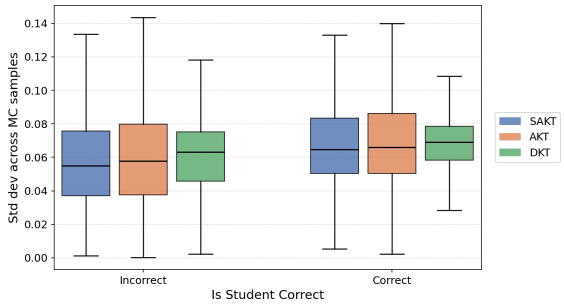

Figure 6: Box plot of each model's prediction standard deviation split into whether the student chose a correct response or an incorrect response.

C.2. Uncertainty Trajectory Across Question Position

While the mean total entropy remains similar across the different architectures (Figure 5), the mean prediction standard deviation displays visible differences (Figure 6). DKT exhibits the largest median spread, while SAKT and AKT show similar but smaller spreads. This cross-architecture spread comparison should be interpreted with the caveat that dropout rates differ across models (0.5 for SAKT, 0.2 for AKT and DKT; Table 5); the observed spread ranking therefore partly reflects this configuration choice rather than purely architectural differences. The broader claim that MC-Dropout exposes architecture-specific epistemic content is robust to dropout-rate calibration, but the per-model spread ordering is not. Figure 7 plots entropy against question position. DKT registers a higher entropy on the very first question than the attention-based models, suggesting that attention-based architectures more easily learn an initial question bias from the training data. A repeating

oscillation emerges in the trajectory, periodic with the five-question quiz structure (Figure 9 shows the matching question-difficulty oscillation); we examine the entropy–difficulty relationship directly in Appendix C.3. Figure 8 presents the same trajectory using prediction standard deviation. The spread for DKT increases over the first 20 questions before stabilising, whereas SAKT and AKT decrease as more history accumulates. The attention-based models grow more confident over time, while the LSTM-based model does not.

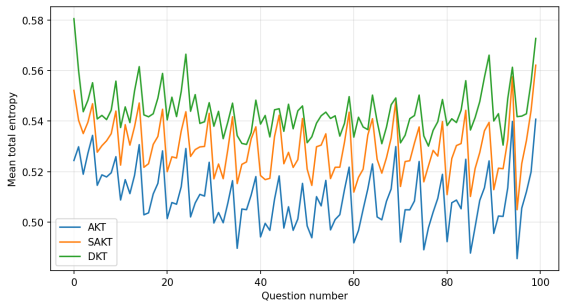

Figure 7: Each model's predictive total entropy against the question number currently being predicted.

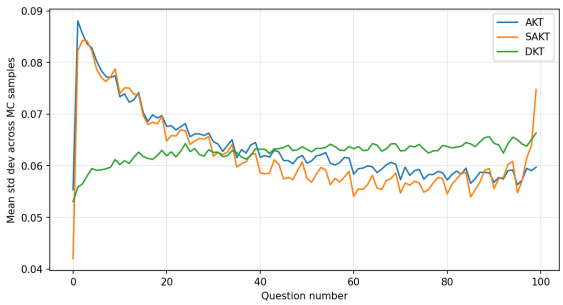

Figure 8: Each model's prediction standard deviation against the question number currently being predicted.

C.3. Correlation with Question Difficulty

Figure 9 illustrates the mean question difficulty against question position, exhibiting a sawtooth pattern that reflects the within-quiz progression. Figure 10 reports a moderately strong negative correlation between mean predictive entropy and question difficulty (Pearson $r = -0.62$, $p < 10^{-11}$). The negative sign is a feature of binary correctness prediction under class imbalance: on harder questions the model can confidently predict failure ($p_{\text{correct}} \to 0$, low entropy), while on easier questions it commits to a moderately confident correct prediction with entropy nearer the binary maximum. The same mechanism produces the slightly higher median entropy on student-correct cases visible in Figure 5; both reflect the same underlying calibration of the binary classifier rather than a failure of the uncertainty signal itself, which remains useful for flagging *model* errors (Figure 4).

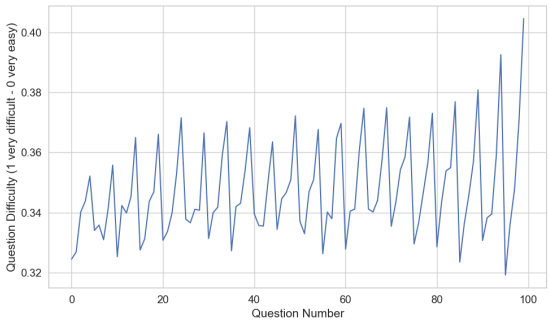

Figure 9: Comparison of question difficulty against the question number that would be predicted by a model.

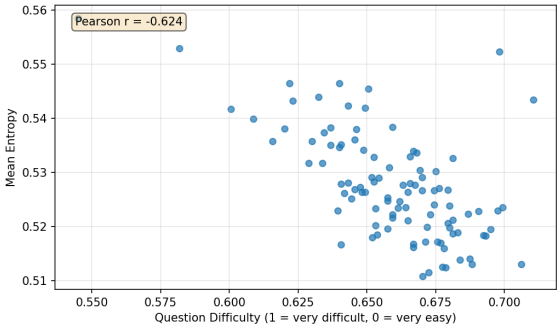

Figure 10: Correlation between the model predictive entropy and the question difficulty.

Appendix D. Subgroup Fairness Beyond Student Ability

The paper's fairness analysis (Figure 2a) shows that abstention rate is close to the 20% uniform baseline across student-ability quartiles. Eedi's user metadata also records gender and year-group; we extend the fairness check to these subgroups in Figure 11.

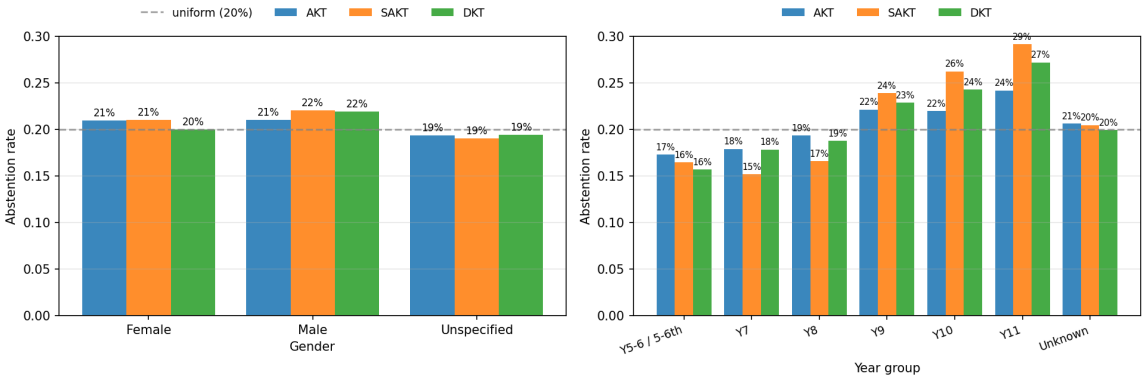

Figure 11: Abstention rate at $c = 0.80$ broken down by self-reported gender (left) and country-year group (right). Gender-conditioned rates fall within 19–22% across all three architectures, very close to the 20% uniform line. Year-group rates trend upward with student age (15–18% for Year 5–8 vs. 22–29% for Year 9–11), showing that the abstention is not evenly distributed across cohort age but does not single out any specific group for over-deferral.

Gender: female, male, and unspecified-gender students fall within 19–22% abstention across all three architectures. The selective layer treats the three groups equivalently within sampling noise.

Year group: abstention rises monotonically with student age, from 15–18% on Year 5–8 cohorts to 22–29% on Year 9–11. This trend is consistent across architectures and most likely reflects question difficulty: older students are exposed to objectively harder mathematics content, on which the model has more uncertain predictions. We do not interpret this as a fairness violation, since no specific group is being over-deferred relative to its underlying difficulty profile. However, a deployment system that wished to enforce a strict per-cohort abstention budget would need to apply per-group thresholds rather than a single global c. Per-group calibration of the operating point is left to future work.

Appendix E. Variance Decomposition of MC-Dropout Epistemic Uncertainty

Question being asked. The variance decomposition is a falsification test for the redundancy hypothesis: is MC-Dropout's signal recoverable from publicly observable, model-free features of the (student, question) pair?
If yes, MC-Dropout is a complicated re-implementation of cheaper proxies and a deployment system could substitute one for the other; if no, MC-Dropout surfaces signal those proxies cannot reconstruct.

Target variable. For each of the ∼110,000 last-70 prediction targets per architecture, we compute the model's *epistemic* uncertainty using the BALD decomposition (Houlsby et al., 2011):

$$\underbrace{H\big[\,\mathbb{E}_w\, p(y \mid x, w)\,\big]}_{\text{total}} \;-\; \underbrace{\mathbb{E}_w\, H\big[\,p(y \mid x, w)\,\big]}_{\text{aleatoric}} \;=\; \underbrace{I(y; w \mid x)}_{\text{epistemic}},$$

where the expectation is approximated over $M = 100$ stochastic dropout-active forward passes. The aleatoric component is subtracted out so the regression target is purely the model's *reducible* uncertainty over its own learned weights.

The decomposition uses BALD epistemic but the rest of the paper uses total entropy. Throughout the paper we use total predictive entropy and prediction variance as the primary uncertainty signals, since these are the quantities the selective-prediction layer actually sorts on and the descriptive figures most directly visualise. The information-theoretic decomposition of total predictive entropy into aleatoric and epistemic components is known to have identifiability and additivity caveats: the additive split does not always correspond to a clean separation of irreducible-versus-reducible uncertainty, and the empirical estimates inherit instabilities from the dropout posterior approximation (Wimmer et al., 2023; Valdenegro-Toro and Mori, 2022). We therefore avoid leaning on the decomposition where it is not strictly necessary, and use total entropy / variance as the operational signals.

For the variance decomposition specifically, however, the question being asked requires isolating the part of the signal that classical psychometric proxies cannot recover. Proxies like IRT outcome ambiguity ($H(p_{\text{IRT}})$) are aleatoric-flavoured by construction: they only see student ability and question difficulty, never the model's representational state. Regressing total entropy on these proxies would therefore partly be regressing the aleatoric component on its own surrogate, inflating the explained R^2 without telling us whether the architecture's epistemic content itself is recoverable. Subtracting the aleatoric estimate via BALD gives

a more honest test, even if the subtraction is imperfect. As a robustness check, we reran the linear decomposition with total predictive entropy as the regression target and found the qualitative conclusion intact: classical psychometric proxies still leave 60%–64% of the signal unexplained across all three architectures. After the aleatoric component is subtracted out, the same linear regression on BALD epistemic uncertainty leaves 96%–99% unexplained, falling to 77%–90% only when we permit a non-linear regressor (see below). The architecture-specific dominance therefore does not depend on either the BALD subtraction or the linear functional form. It is amplified by both, but already present without them.

Predictors. Each prediction is annotated with classical psychometric features that a deployment system could compute without running the model:

- **Question difficulty**: empirical correctness rate of question q across the validation cohort.
- **Student ability**: per-user accuracy across that user's full 100-question sequence.
- **IRT outcome ambiguity**: $H(p_{\text{IRT}})$, the Shannon entropy of the 2PL prediction $p_{\text{IRT}}(u, q) = \sigma(a_q(\theta_u - b_q))$, with θ_u fit on the first 30 questions.
- **Curriculum coverage**: four binary indicators (one per curriculum level) for whether the target question's subject / topic / subtopic / construct appears in the same student's first-30 context.

Procedure. We fit a sequence of ordinary least-squares linear regressions of epistemic uncertainty on *nested* predictor sets, computing the coefficient of determination R^2 at each step. The marginal R^2 of the k-th predictor is the additional fraction of variance in epistemic uncertainty linearly explained by adding that predictor on top of those already in the model; the cumulative R^2 at step k is the total fraction jointly explained by the first k predictors. The unexplained residual $1 - R^2_{\text{full}}$ is therefore signal that no linear combination of these proxies captures.

Interpretation of the result. Figure 12 reports cumulative R^2 at each step, for each architecture. The full classical psychometric stack jointly explains 2.4% (AKT), 3.8% (SAKT), and 1.1% (DKT) of epistemic uncertainty under linear modeling. Because each marginal contribution is small, no single predictor is doing the bulk of the work either: this is not the case where one proxy already captures everything and the others add nothing redundant.

Non-linear robustness check. A reviewer might object that linear regression is too conservative: perhaps the same predictors, combined non-linearly, capture much more. To bound this we re-fit the predictor stack as input to a 5-fold cross-validated random forest (`n_estimators=100`, `min_samples_leaf=20`); this is a much more flexible model class but uses identical features and the cross-validation prevents overfitting from inflating R^2. Cross-validated R^2 values are 10.5% (AKT), 9.8% (SAKT), and 23.2% (DKT), leaving 89.5%, 90.2%, and 76.8% unexplained respectively. The non-linear gap-closing is largest for DKT (the LSTM-based architecture), suggesting its epistemic uncertainty has more recoverable interactional structure than the attention-based architectures, but a substantial majority of the signal remains architecture-specific in every case. The model's epistemic

uncertainty therefore varies on dimensions that are not just linearly orthogonal to the classical psychometric features, but largely orthogonal at any reasonable model class: that variation is what MC-Dropout's stochastic forward passes recover.

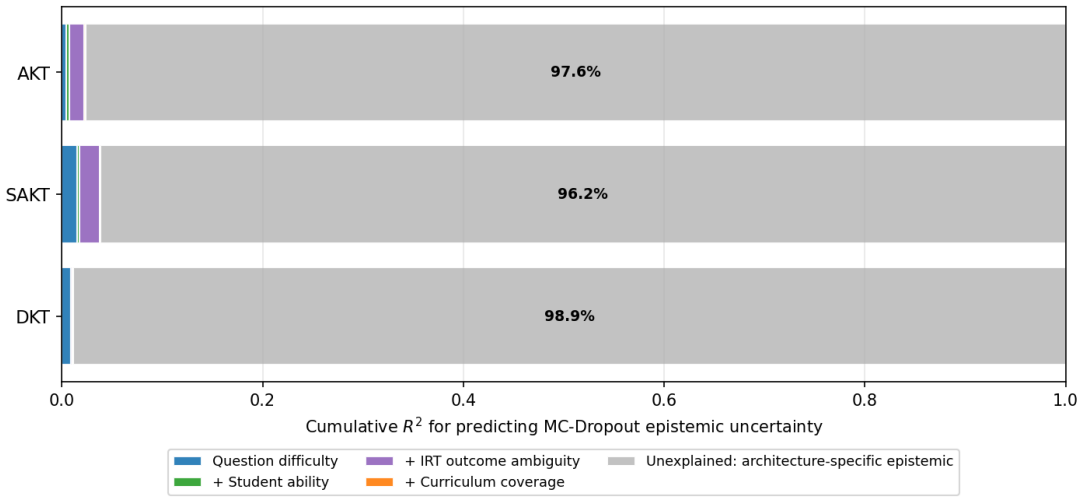

Figure 12: Variance decomposition of MC-Dropout epistemic uncertainty (BALD), under linear modeling. Cumulative R^2 from question difficulty, student ability, IRT outcome ambiguity, and curriculum coverage at all four granularities tops out at 3.8% across all three architectures. A non-linear robustness check (5-fold cross-validation random forest, see text) raises explained R^2 to 9.8–23.2%, leaving 76.8–90.2% as architecture-specific content that classical psychometric proxies cannot recover even at a much richer model class.