Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Enhancing Neural Cognitive Diagnosis: Q-Matrix Recovery, Task Distraction Sensitivity, and Dynamic Attention Modeling from Synthetic Data

Judit Tevesz

teveszjudit@edu.bme.hu

Budapest University of Technology and Economics

Bertalan Forstner

Budapest University of Technology and Economics

Article

Keywords: Neural Network, Cognitive Diagnosis, Deep-Learning, Synthetic Data Generation

Posted Date: March 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-7618651/v1

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

Additional Declarations: No competing interests reported.

Enhancing Neural Cognitive Diagnosis: Q-Matrix Recovery, Task Distraction Sensitivity, and Dynamic Attention Modeling from Synthetic Data

Judit Tevesz, teveszjudit@edu.bme.hu

Dr. Bertalan Forstner, forstner.bertalan@vik.bme.hu

Department of Automation and Applied Informatics
Budapest University of Technology and Economics

Budapest, Hungary

Abstract

Neural Cognitive Diagnosis Models have shown potential for inferring fine-grained student skill profiles. We extend NeuroCDM to enhance its expressiveness and applicability. First, we address the limitations of idealized Q-matrices: enabling the model to operate under incomplete skill-task mappings, improving robustness to real-world instructional noise. Second, we introduce a task-level parameter (δ) capturing the susceptibility of tasks to performance decline under transient attention difficulties - learned from task features via underlying γ parameters. Third, we model student-attention dynamically through a latent parameter φ capturing temporal variations in focus during task execution. To support training without costly human-labeled data, a validated simulation framework is used that generates realistic assessment datasets. NeuroCDM, trained on this synthetic data, accurately recovers 62,3%-100% of the omitted skill dependencies from the Q-matrix, with less than 20% deviation from the original values, and achieves a mean absolute error (MAE) ranging from 0.0555 to 0.1156. We also estimated the γ parameters with MAE below 0.15 for each. Estimations of the ϕ parameter had MAE below 0.015, furthermore a binary AD prediction was made based on performance - the results are also remarkably correct. Together, these contributions advance the scalability and diagnostic power of neural CDMs in educational contexts.

Keywords: Neural Network, Cognitive Diagnosis, Deep-Learning, Synthetic Data Generation

Introduction

Cognitive diagnosis is a growing research area focused on identifying individuals' specific cognitive strengths and weaknesses, with applications in education, psychology, healthcare, and beyond. A central task in this field is estimating the probability that a student can solve a given task, based on their underlying cognitive abilities and the skills required by that task.

Füstös and Forstner [5] proposed a synthetic data generation framework based on multidimensional Item Response Theory (mIRT) to simulate realistic student-task interaction data across multiple cognitive dimensions. This approach allows for controlled experimentation and model evaluation without the cost of collecting real-world assessment data.

Building on this foundation, in [13] this framework was used for estimating student and exercise parameters using a neural network. Each student is represented by a latent ability vector (θ) $\in N(\mu = 1, \sigma = 0.15)$), capturing proficiency across multiple skills. Tasks are defined by a skill-weight vector U(0,1) and a required-skill-level vector ($\mathcal{N}(1, 0.15)$, indicating which abilities are involved and how demanding the task is for each. This modeling setup ensures a nuanced representation of task difficulty and student ability. In the aforementioned work of [13] the NeuroCDM model was used, that is a deep learning-based cognitive diagnosis framework. The architecture of the model is described in details in [15]. In a comprehensive evaluation across various configurations of students (N), tasks (M), and skills (K), the model achieved promising results:

- θ-estimation errors: absolute mean difference ranging from 0.0996 to 0.167
- β-estimation errors: absolute mean difference ranging from 0.088 to 0.124

These low error rates were achieved after only 10 training epochs, demonstrating the model's efficiency. Importantly, the number of exercises per student was shown to have limited impact on performance, suggesting the method's robustness across different testing scenarios.

These results demonstrate that combining synthetic data generation [5] with deep learning-based inference models like NeuroCDM enables accurate, scalable, and cost-effective cognitive diagnosis without relying on large-scale real-world datasets.

The aforementioned researches used only the students' and tasks' identifiers besides the Q-matrix and the successfulness of the task-solving as input. We extended this approach by incorporating the model's predictions of an incomplete or errored Q-matrix, which represents the knowledge relevance of different cognitive skills required for each task. This extension allows the model to handle more complex scenarios where the Q-matrix may have missing or incorrect information, thus improving its robustness in real-world settings. Additionally, we introduced two new parameters: the task's attention-disorder triggering weight (denoted as δ) and the attention level of students at each moment (denoted as ϕ). The δ parameter represents the extent to which a task is sensitive to attention deficits, such as those caused by ADHD or other transient disturbances, with higher δ values indicating tasks more affected by attention lapses. Tracing back to task-specific metrics (e.g., long instruction, monotonicity, visual complexity) it can be defined as a linear combination of these factors, where the γ parameters—currently calibrated by psychometricians—can be learned during model training along with tasks' dependencies and student abilities. The ϕ parameter tracks the student's attention during task execution, offering a dynamic view of how student concentration fluctuates over time. By integrating these parameters, we improved the model's ability to predict student performance more accurately, especially where attention and task-related challenges are influential.

2 Methodology

2.1. Estimating the existence of skill-dependencies and estimating their values in an incomplete Q-matrix

Empirical results demonstrate that the model - in the absence of prior information or in the presence of an incomplete/inaccurate Q-matrix - is able to reconstruct not only the binary presence of skill dependencies, but also their relative magnitudes based solely on learner response data.

The Q-matrix is denoted as $Q = [Q_{ij}] \in \{0, 1\}^{M \times K}$, where: $Q_{ij} = 1$, if exercise e_i relates to knowledge concept k_j, 0 otherwise. However, manually labeled Q-matrix may be deficient because of inevitable errors and subjective bias. Traditionally, Q-matrices are constructed by subject-matter experts based on their understanding of the skills needed for each item. As also noted by Liu, Xu, and Ying [8] the expert-driven process is highly susceptible to human error, including misjudgments about which attributes are actually required, inconsistencies across experts, and oversights in representing complex cognitive processes. A mis-specified Q-matrix undermines the validity and diagnostic power of the assessment [2]. Therefore, while expert input remains valuable, increasing attention has been given to data-driven methods that aim to refine or validate the Q-matrix empirically, reducing the risk of error and improving the alignment between test items and the cognitive skills they aim to measure.
In this section we test whether the model is able to find the underlying connections that were not mentioned in the Q-matrix. We aim to observe not only the presence of a knowledge-skill but whether the model can predict the importance of the skill in the task. We manually drop random tasks' knowledge-dependencies. This aims to observe whether the model is able to realize if a task is dependent from an ability even though it was not mentioned by psychometrics. E.g.: a task that is categorized by "Equation" might cause groundless difficulties to someone who has no trouble in solving equations, but with fractions/exponentiation, and such operations are included in the problem.

2.1.1. Data generation

Synthetic data is generated the same way as mentioned in [5]. After data generation and creating the Q-matrix, task IDs are randomly selected, and among the dependencies the index of the knowledge-skills to be removed are also randomly selected. For the current setup we modified 40 rows from the Q-matrix. At each task randomly one or two skill-dependencies were removed, if minimum one dependency remained afterwards – not to leave any task without dependence. It should be noted that randomly removing dependencies does not fully capture the nature of real-world expert errors. In practice, human annotation errors are often systematic rather than random — arising from cognitive biases, domain-specific assumptions, or ambiguous task formulations. Future works aim to simulate more realistic annotation deficiencies, including structured or biased omissions, to better reflect the challenges faced in practical psychometric applications.

2.2. Extending the dataset with γ parameters for tasks' "distraction sensitivity" characteristics

In this section we aim to show that the model is capable of estimating the weights of a task's "distraction sensitivity" characteristics. The model fundamentally operates with static abilities but can be extended to also consider transient or situational effects—like attention deficit, stress, or fatigue.
To simulate a transient distraction, noise can be added to the equation that gives the probability of a student solving a given task correctly:

$$P(Y_{ij} = 1) = \sigma\left(\alpha_j \cdot \left(\sum w_{jk} \cdot \theta_{ik}\right) - \beta_j - \delta_j \cdot z_{ij}\right)$$

Where z_{ij} is a noise term that is different for each person-task pair. This can represent situational effects—e.g. how focused the student was at that moment.

In educational/psychological modelling (e.g., NCDM, MIRT, or attention-aware CDMs), δ is often used to model student-level bias or attention modulation over tasks.

Formally, δ could represent:

- A global bias in how a student approaches tasks (e.g., risk-taking, guessing),
- An effort/engagement trait that scales task performance independently of knowledge,
- A regularization or gating mechanism for inconsistent behavior —highly relevant for modelling AD.

In our extension, δ parameter represents the extent to which a particular task is sensitive to ADHD or other transient (temporary) attention deficits. If a task is particularly "transient-prone", then δ receives a higher value. This indicates that for such tasks, temporary attention disturbances have significant effect on the likelihood of success.
If task-specific metrics are available that indicate sensitivity to transient disturbances then the parameter can be defined as a linear combination as follows:

$$\delta_j=\gamma_0+\gamma_1\cdot\text{instruction_length}_j+\gamma_2\cdot\text{monotonicity}_j+\gamma_3\cdot\text{visual_complexity}_j+\dots$$

Currently, the γ parameters are calibrated by psychometricians, but according to this research and our empirical proof by testing, these parameters can be learned during model training—along with the task ability-dependencies and student abilities. It is important to note that the interpretability and identifiability of the γ parameters—when δ is modelled as a linear combination of task-level features—assumes low multicollinearity between them. This means in practice, that features should be approximately orthogonal. While our current approach is designed with this in mind, future extensions may explore feature decorrelation techniques or regularization strategies to ensure parameter stability and interpretability.

2.2.1. Data generation

Our algorithm extends [5]: exercise data is augmented with features that have the potential to induce attention deficits. The extent to which these features are present in a specific task is quantified using a Likert scale ranging from 1 to 5. Assume there are G attention-features, represented as $A=\{a_1, a_2 \dots a_G\}$.
In the specific example referenced above: G=3, as the task includes attention-related features like: lengthy instruction, requirement for monotonic processing, and high visual complexity. This set of features can be extended further.
The risk-of-attention-deficit is measured by δ for each task. The effects of such unique features on triggering attention deficit are defined by γ values: γ_0 is a default value that is independent from the features as every exercise is attention-sensible in some ways. The Likert-value vector of the features is multiplied by a weight vector. Values are sampled from $\mathcal{N}(\mu=0.4\ \sigma=0.15)$, similarly to γ_0 that is added to the product.
A mean of 0.4 situates this baseline sensitivity at a moderate level, neither negligible nor overwhelming. Almost all cognitive tasks involve some degree of attention demand, even if they don't score high on monotonicity, instruction length, or visual complexity. μ=0.4 for γ weights suggests that on average, each feature contributes moderately to the attention risk (δ). This aligns with the Likert scale-design: with moderate weights, the aggregated δ values will typically fall in a range that is behaviorally interpretable (e.g., between 0.5 and 2.5), without producing unreasonably high or low values. A standard deviation of 0.15 allows for sufficient variation in γ values (about 95% of values falling between 0.1 and 0.7), enabling tasks to be more or less sensitive to certain features without causing extreme

parameter instability. This balances model expressiveness with parameter regularity, especially important in psychometric contexts where identifiability and interpretability are critical.

Additional noise is added to δ and their product is subtracted from the expression (as attention-deficit hinders the probability of correct solution), thus the probability of solving a task is calculated the following way:

$$P(Y_{ij} = 1) = \sigma\left(\alpha_j \cdot \left(\sum w_{jk} \cdot \theta_{ik}\right) - \beta_j - \delta_j \cdot z_{ij}\right)$$

We aim to show that a neural network is able to learn and efficiently predict the attention-triggering feature sensitivity of the tasks.

3.3.2. Model Structure

We extended the NCDM model [15] with a gamma_weight layer and a gamma_bias (referred to as $gamma_0$).

3.3.2.1. Gamma weights

Embedding layer: shape: *(M x G)*
This is a learnable embedding that maps each exercise (identified by its ID) to a vector of weights corresponding to attention features. For an exercise *e*, gamma_weights[*e*] is a vector of length G (default: G=3) that determines how important each attention feature is when computing the delta adjustment.

3.3.2.2. Constant value $gamma_0$

Embedding layer: shape: *(M, 1)*
$gamma_0$ is a scalar bias term associated with each exercise. Serves as a baseline adjustment to the prediction for each exercise, before considering any attention-based correction.

The term δ is a dynamic bias for the prediction, customized per exercise and per input sample via the attention features. gamma_weights *(M x G)* * attention_features *(G x 1)* perform element-wise multiplication between the learned weights and the input attention features. The sum reduces that to a scalar per sample *(M x 1).* Adding γ_0 completes the calculation of delta.

3.4. Extension of the dataset for estimating the students' attention-state by introducing ϕ variable and predicting probability of the presence of attention-disorder (AD)

The model is capable of estimating whether a student potentially has attention difficulties based on performance (i.e., the estimated probability of success). In addition to this binary classification, the model can also infer the student's attentional state during the solution of a specific task, taking this information into account separately from the student's underlying abilities.

The equation that shows the probability of successfully completing a task in this case:

$$P(Y_{ij} = 1) = \sigma\left(\phi_i \cdot \alpha_j \cdot \left(\sum_{k=1}^{K} w_{jk} \cdot \theta_{ik}\right) - \beta_j\right)$$

3.4.1. Data generation

The algorithm extends [5]. Additionally, each student is randomly assigned a binary AD label:

- 0 → does not have AD (60% of data)
- 1 → has AD (40%)

This simulates a set of students where AD is overrepresented compared to real-world prevalence (to ensure enough AD cases for modelling).

For each person, a list of M attention/engagement values (ϕ) is generated (ϕ-array with shape: N x K). The values are sampled from a normal distribution, regarding the following:

- If the person has an attention-disorder, ϕ values are drawn from $\mathcal{N}(0.6, 0.2)$: Lower mean, higher variance: represents lower, inconsistent attention. Barkley [1] and Sonuga-Barke [12] highlight that individuals with ADHD often exhibit significant difficulties in sustaining attention over time. Attention is described as less stable and prone to higher variability, particularly when tasks require prolonged concentration or the inhibition of distractions. These individuals tend to be more impulsive and often display greater attention fluctuations, which can result in inconsistent performance across tasks.
- If the person does not have AD, ϕ values are drawn from $\mathcal{N}(0.8, 0.05)$: Higher mean, tighter spread represents more stable and higher attention. The Sonuga-Barke study [12] agrees that ADHD individuals have increased attention variability, which justifies the use of a normal distribution with higher variance for AD, in contrast to non-AD individuals who are more likely to have stable attention with lower variance.

3.4.2. Model structure

We extended the NCDM model with an ad_head layer for the binary prediction of whether the student has AD and also with a φ-matrix.

3.4.2.1. AD-Head

Purpose: To perform binary classification for each student, based on their latent knowledge state as learned by the model. Let:

- $\theta_j \in R^d$: latent embedding vector for student *i* (from the model's student-embedding)
- f_{AD}: $R^d \rightarrow [0,1]$: prediction head mapping from student embedding to AD probability.

Then the AD prediction is given by:

$$\widehat{a_n} = \sigma\left(w_{ad}^T \cdot \theta_i + b\right)$$

Where:

- σ is the sigmoid-activation

- $w_{ad} \in R^d$, b∈R: parameters of the linear layer (ad_head).

Interpretation**:**
The model learns to associate certain patterns in a student's knowledge vector (e.g. inconsistencies or low engagement) with AD. Applying a sigmoid function on the output of the ad_head layer outputs a scalar between 0 and 1 for each student, representing their predicted tendency towards attention difficulties. Resulting in a binary classification, supervised by true AD labels during training. The prediction is part of a multi-task learning setup: one task is to predict exercise correctness; the other is to estimate AD likelihood.

3.4.2.2. Phi Table (Attention Weights)

It captures the student-specific attention or engagement level when interacting with each exercise. It dynamically scales the influence of the difference between student ability and exercise difficulty. Let:

- i ∈ [0, N−1]: student index.
- j ∈ [0, M−1]: exercise index.
- $\phi_{ij} \in [0,1]$: learned attention or modulation scalar for student *i*'s interaction with exercise *j*
- The lookup index is computed as: $idx_{ij}=i\cdot M + j$

Then:

$$\phi_{ij} = \sigma(Embedding[idx_{ij}])$$

The model's phi_table is an embedding table of size (N·M, 1), where each scalar is passed through a sigmoid to ensure it's between 0 and 1.

$$input_e=\phi_{ij}\cdot h_j\cdot(\Theta_i -\beta_j)\odot x_j$$

where

- h_j is the exercise-discrimination value
- Θ_i is a latent embedding vector for student i
- β_j is a knowledge requirement vector for item j
- x_j is a binary indicator of knowledge concepts related to item j
- $\odot$: element-wise multiplication.

AD-Head: Captures persistent, student-level cognitive traits across exercises.

φ-Layer: Captures transient, student–exercise-specific factors like attention or effort —might be inconsistent in AD students.

Together, these components let the model both predict behavior and diagnose learning difficulties, making it suitable for cognitive diagnostic modelling with behavioral insights.

As it is a multi-task learning where the model learns better representations by trying to jointly predict both:

1. Whether the student answers correctly (main output of the model)
2. Whether the student has AD (auxiliary output of the model).

4 Model training, obtaining embedded values

The model was trained similarly as described in [13] with the same parameters.

4.1 Modified Q – matrix

Model is trained on the modified Q-matrix, then the embedding values are transformed and scaled back to the original distribution. The critical tasks (which were modified), specifically the left-out indices are observed and compared with the original values: whether the model can predict not only the dependency, but the weight of the skill for the task.

4.2. Gamma values

The model is trained on only student ID, exercise ID, result (0/1), the Likert-scale values and the additional noise. The model learns the student skills and exercise skill-dependency values that are stored in separate embedding layers as well as the learnt values for the γs (2 layers: 1 for γ_0, and 1 for $\gamma_{1\text{-}3}$).

Once the model has been trained, we retrieve the student ability vectors and the item skill-dependency vectors but mainly focusing on the two γ embedding layer produced by the encoder. The NCDM model's embeddings that stand for the γ-vector and γ_0 value have to be rescaled to $\mathcal{N}(0.4, 0.15)$

4.4 Attention-disorder-estimation and quantifying attention (ϕ)

Model was trained by providing attention-disorder labels (0/1) during training, that is reasonable, as the goal is not just to "predict AD" as a classifier — for letting the model learn internal features reflecting attention stability, which in turn improve prediction of task performance. AD prediction guides the student embedding (stat_emb) to encode attention-related traits. These embeddings are then used to modulate task success predictions — potentially improving personalization and robustness.

After the model training, we retrieve the AD-head and ϕ embedding layer. The ϕ-values have to be rescaled depending on the model's prediction of whether the person is suffering from an attention disorder or not. Therefore, if according to the model the person is likely to have an ADHD-like attention-disorder, the ϕ values of this student are rescaled to $\mathcal{N}(0.6, 0.2)$, otherwise to $\mathcal{N}(0.8, 0.05)$. Φ values of each individual are then compared to the original ones.

5 Evaluation and Experimental Setup Summary

To assess the model's ability to recover true latent parameters, evaluation metrics included Mean Absolute Error (MAE), Mean Squared Error (MSE). Errors were contextualized following common practices in psychometrics and IQ-tests ($\mathcal{N}(1, 0.15)$), and per-skill error statistics (average and maximum MAE/MSE) were also computed for interpretability. Results showed consistently low error values, supporting the reliability of the NeuroCDM in estimating latent parameters. Given the limited added value of MSE and consistent error magnitudes across other parameters (γ and ϕ), subsequent evaluations focused on MAE for clarity.

[13] validates the NCDM model's estimation on the Θ and β values, therefore based on these results we applied the same experimental setup. This configuration was designed to evaluate model scalability and asymptotic stability—specifically, whether error rates for missing βs, γ, and ϕ converge as data size increases. Fixing M isolated the effect of sample size, enabling analysis of whether a critical threshold exists beyond which more data yields diminishing accuracy gains. Together, these setups comprehensively demonstrate the model's generalizability, consistency, and practical relevance across diverse educational settings.

[13] validates the NCDM model's estimation on the Θ and β values, therefore based on these results we applied the same setup for the aforementioned topics.

6. Result Analysis

6.1. Estimating the values of the missing abilities in the Q-matrix

6.1.1. Results of estimating the β values of the missing abilities in the Q-matrix

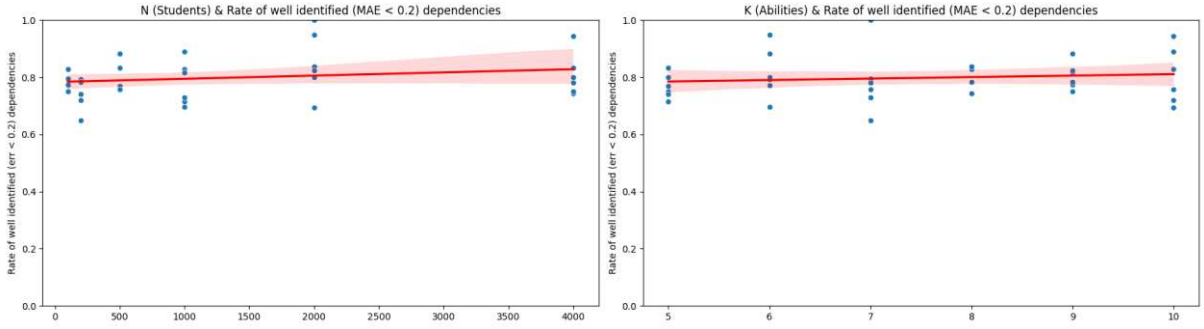


Figure 1, 2: Rate of the number of abilities, where model's β compared to the original one is less than 0.2

We aimed to observe the rate of identified missing dependencies and the MAEs of model estimations of these. The number of exercises was taken as constant 50. We can observe a slight increase on both Figure 1 and Figure 2, therefore increasing the number of students as well as increasing the number of measured abilities resulted in higher rates of preferable predictions. The rate of "well-predicted" skills was minimum 0.6486 and at most 1, meaning that in some setups the model could identify all of the missing elements of the Q-matrix; while the average of absolute differences of the weights was in the range of 0.0629 to 0.1068: with a 95% confidence interval [0.0814, 0.0880]. This can be observed in Figure 3. We observe similar performance than the one described in [15]: the performance is less constant when having only a small dataset and is becoming stable as we increase the size of the dataset – proving the estimating power of NNs with large datasets.

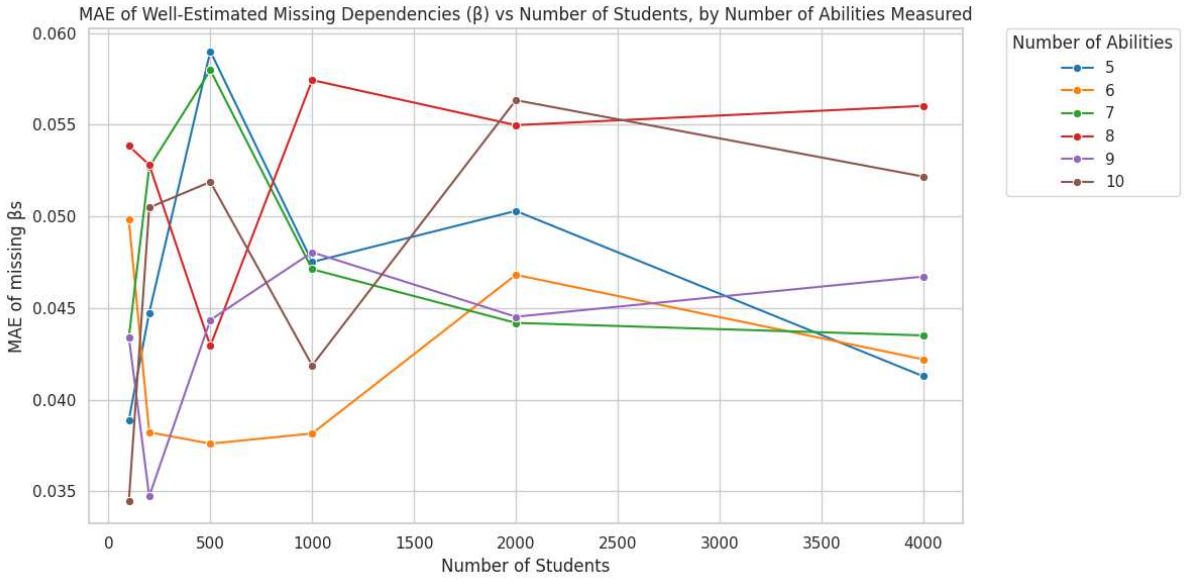


Figure 3: MAE of the model's well-estimated dependencies' βs

6.2. Estimating the γ values of the tasks

6.2.1. Results of estimating the γ values of the tasks

Figure 4, 5, 6, 7 illustrate the performance of the estimation procedure as the number of students increases exponentially. Despite this substantial growth in sample size, the mean absolute error (MAE) for the γ parameters remains consistently low across all conditions. This indicates that the model's estimation accuracy is not significantly affected by the scale of the data, underscoring its ability to generalize well across different sample sizes. The results also show strong convergence behavior, with stable parameter recovery even at the lowest sample sizes and no degradation in performance as the number of students doubles at each step. This consistency suggests that the estimation framework is both scalable and reliable, making it well-suited for practical applications involving large-scale assessments where maintaining accuracy across diverse population sizes is essential.

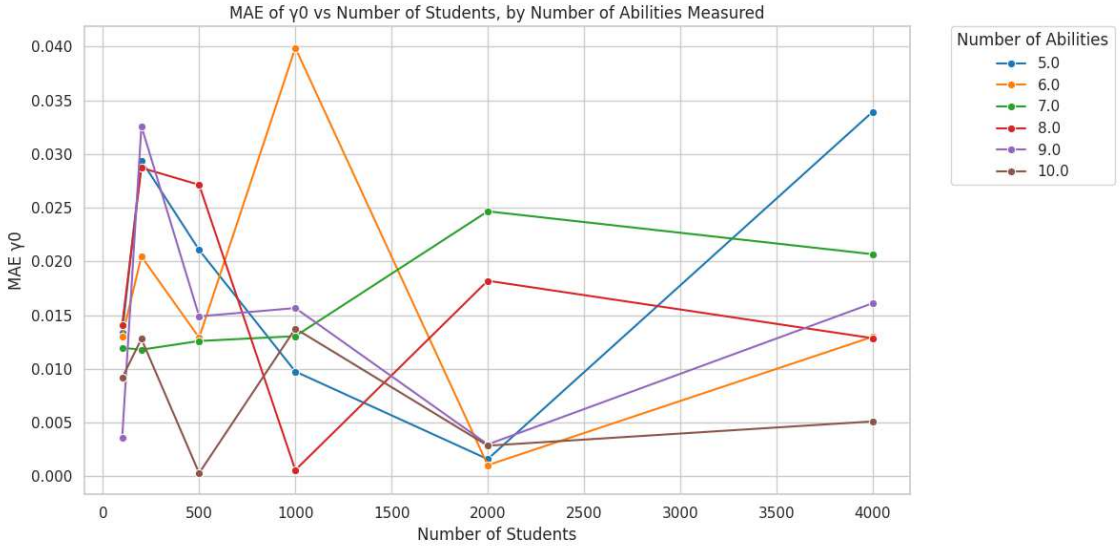


Figure 4: MAE of γ_1

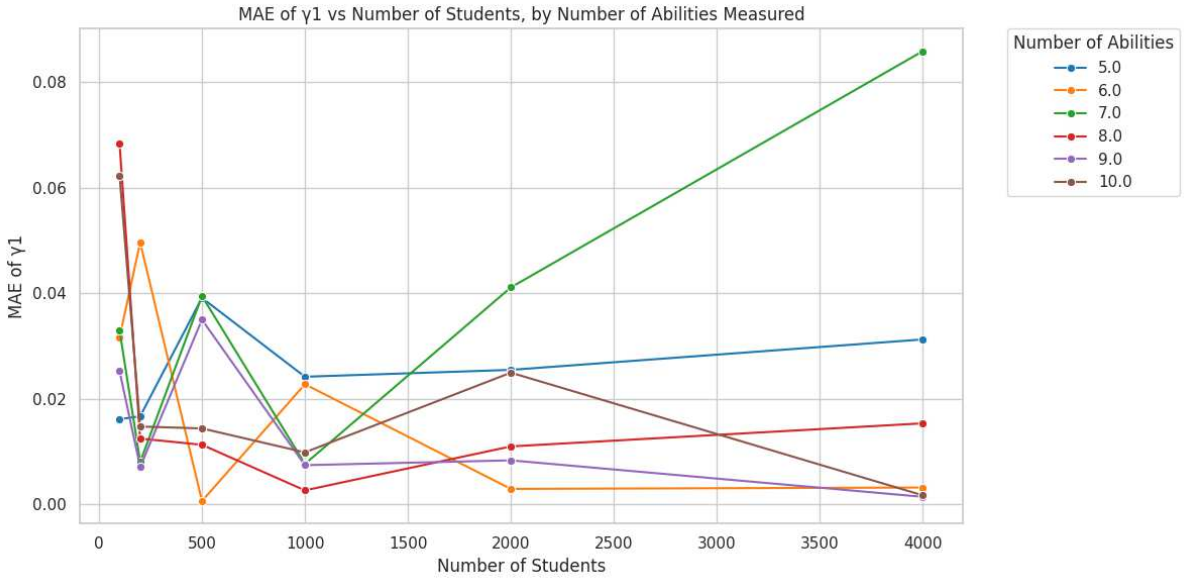


Figure 5: MAE of γ_1

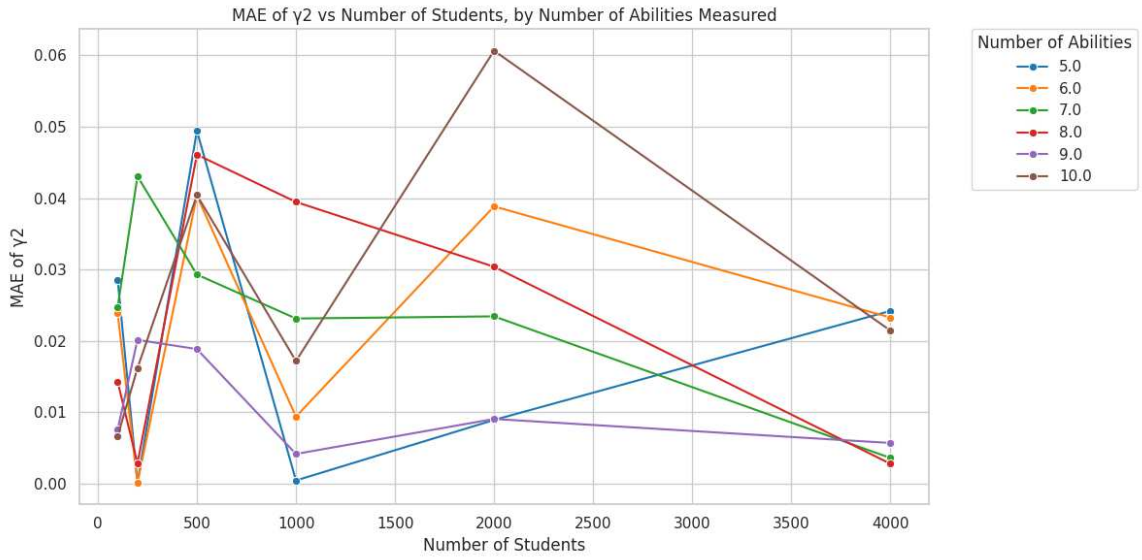


Figure 6: MAE of γ_2

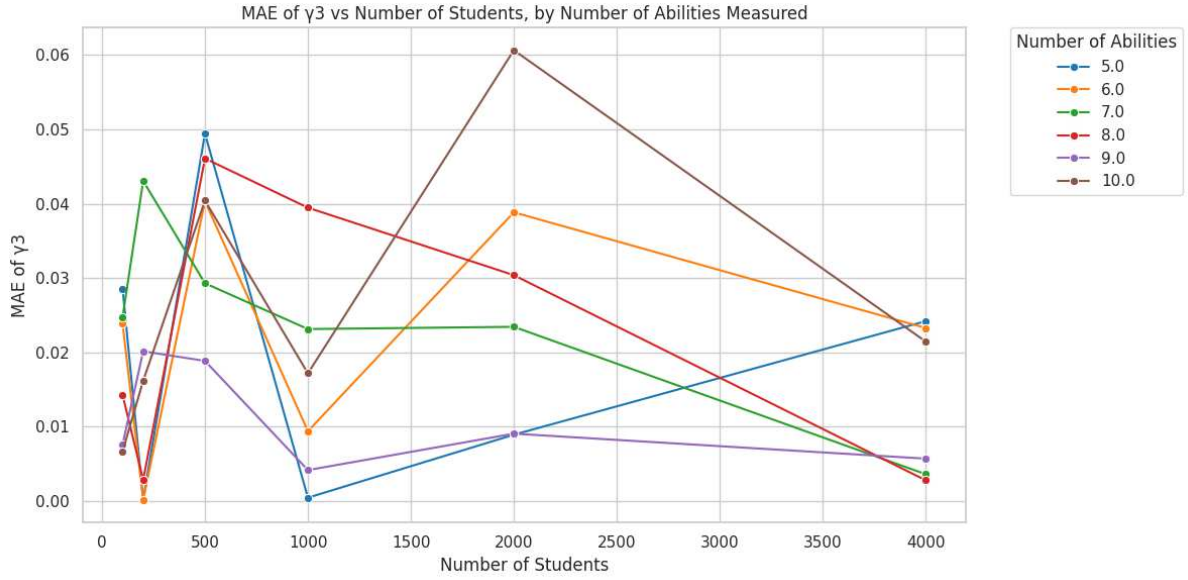


Figure 7: MAE of γ_3

The MAE of γ_0 in this setup was in the range of 0.000286 to 0.039898, for γ_1 the range was 0.000626 to 0.08585, for γ_2 0.0001698 to 0.06649218, and for γ_3 0.00014623 to 0.060651. These are all relatively small differences regarding the original distribution of γs, that is a 0.4 centered normal distribution with standard deviation of 0.15.

The range of MAEs of the γ parameters is slightly wider in this setup than in the previous one, when regarding larger dataset, however, compared to the ground truth 0.4-centered normal distributions, even the MAEs around –0.04 and 0.04 that are the largest in absolute values mean only several or several decimals percentage compared to the real values.

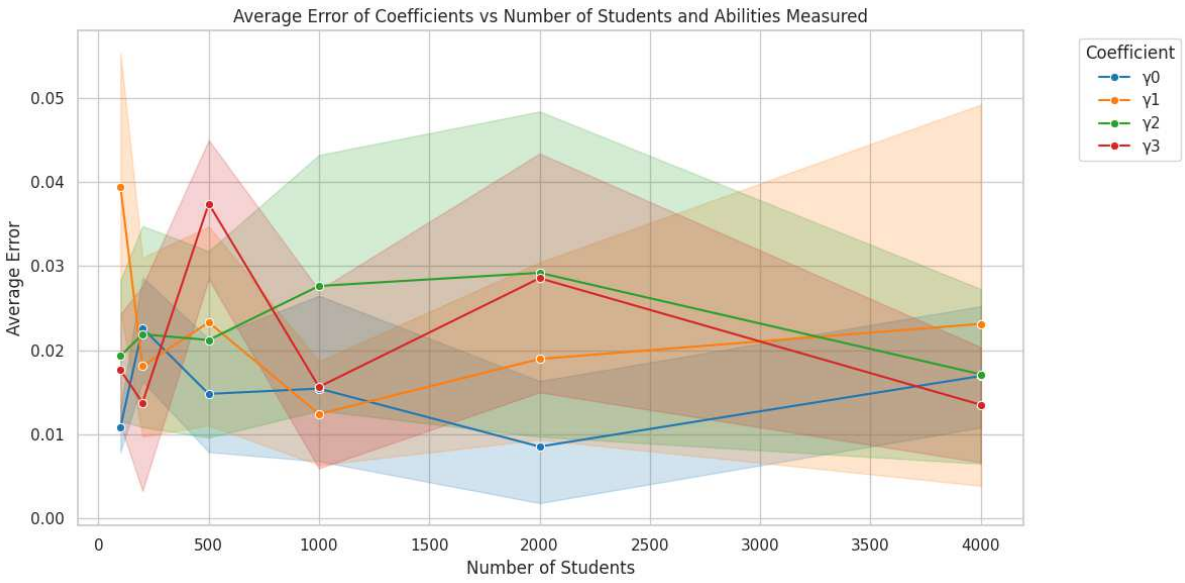


Figure 8: MAE ranges for all γ parameters

Figure 8 illustrates the range of mean absolute errors (MAE) for all γ parameters across the various setups, considering only the increasing number of students and measured abilities. It can be observed that the MAE values remain low across all estimations, indicating that the final estimated value of the task's attention-disorder triggering weight (δ) closely approximates the true value. Consistent with the previous findings the model exhibits larger errors when trained on smaller datasets; however, these errors remain relatively minor compared to the ground truth. Furthermore, the model's performance improves with increasing dataset size and stabilizes thereafter. Although the error range becomes wider at 4000 students, the average error remains acceptably low, suggesting robust estimation performance even at higher data volumes. Further investigations should be done by using a constant number between 2000 – 3000 of students as the MAEs are the lowest for all parameters in this case.

6.3. Estimations of the attention-metrics (φ) of students

6.3.1. Results of the estimations of the attention-metrics (φ) of students

Figure 9 shows that the model also maintains its stable performance in predicting the value of the student's current attention. Despite the increased complexity and larger sample sizes, the model operates with remarkable accuracy. This stability in Φ estimation, regardless of sample size, underscores the model's reliability and robustness in assessing attentional characteristics, making it a valuable tool for both large-scale and individualized diagnostics. The MAE of ϕ predictions can be observed in Figure 9:

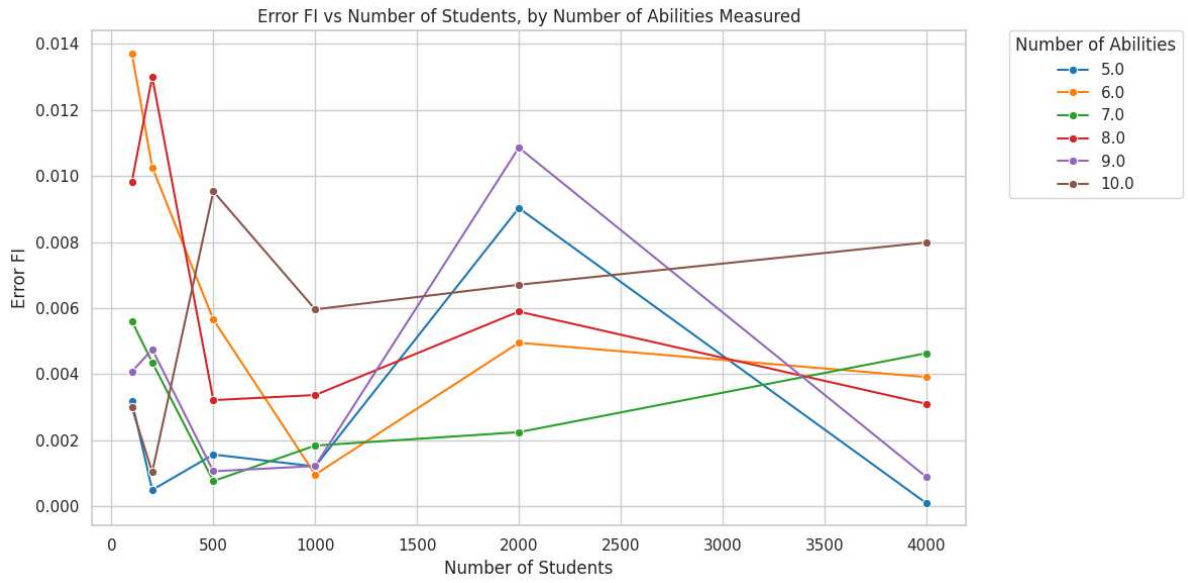


Figure 9: Effect of Number of Students & Abilities on MAE of φ

The MAE of the predicted φ values were in the range of 0.0001026 to 0.0137 with a 95% confidence interval of [0.0035, 0.0059], which is appropriately small compared to the original distribution where the φ values are drawn from ($\mathcal{N}$(0.6, 0.2) and $\mathcal{N}$(0.8, 0.05)).

The extremely low MAE values can appear unusually optimistic. This likely reflects an idealized simulation setup with limited or no stochastic noise. In practical applications, such precision is rarely attainable due to model misspecification, measurement error, and inherent data variability. Therefore, these results should be interpreted with caution, as they may not generalize to real-world scenarios. Future work will aim to incorporate more realistic simulation conditions, including the introduction of stochastic noise, to better reflect the complexities encountered in actual applications.

7 Real life data

We have real-life data containing only students' responses and the Q-matrix. Therefore, the other parameters could not yet be tested.

After validating the approach on synthetic data, we applied the model to real-world datasets: PISA 2012 and TIMSS 2007. Both datasets consist of students solving mathematical exercises, focusing on various cognitive and mathematical skills. In the PISA dataset, 12 exercises and 6 skills were considered—skills such as Mathematical Abstraction, Logical Reasoning, Mathematical Modeling, Intuitive Imagination, Mathematical Operations, and Data Analysis. Most PISA tasks involved three skills. The TIMSS dataset, on the other hand, comprised 24 exercises and 7 skills, covering domains such as Numbers, Geometric Shapes and Measurement, and Data Display, with most tasks involving only 1–2 skills.

For both datasets, the Q-matrix was available, encoding the knowledge dependencies of the tasks. Our goal was to test whether the neural-network-based model could reconstruct the Q-matrix from its embedding layers after training on student response data, as was demonstrated previously with synthetic data. The model successfully identified task–skill dependencies. We considered a skill relevant when its corresponding embedded beta value exceeded a given threshold.

For the PISA dataset, the highest accuracy was obtained at a threshold of 0.6, where a skill was deemed relevant if its beta value exceeded this level. At this setting, we achieved an accuracy of 0.806 and an

F1 score of 0.65. Results remained robust across thresholds between 0.4 and 0.8, confirming that the model reliably identified relevant skills. Since the model can reconstruct a Q-matrix with high precision, we argue that it is suitable for predicting students' skill levels and for estimating the degree of task–skill dependencies—extending beyond the binary information traditionally provided in a Q-matrix.

For the TIMSS dataset, the best accuracy was 0.789, also at a threshold of 0.6. Similar accuracy scores were observed for thresholds in the range of 0.5–0.9. However, the F1 scores were lower (0.32–0.36), reflecting the challenge posed by tasks involving only 1–2 out of 7 possible skills, which makes it harder for the model to precisely distinguish skill relevance.

8 Conclusion

In this paper we demonstrated that a NN approach can effectively incorporate educational prior knowledge into the learning process. By integrating explicit domain knowledge, the model NeuroCDM, improves both interpretability and predictive performance. Experiments on several benchmark datasets show that NeuroCDM consistently outperforms state-of-the-art models.

The main contributions of our work:

- We suggest solutions for correcting the human-labelled exercises that are might be incomplete, in other words that a neural network can precisely estimate the underlying skill-dependencies of the exercises.
- We suggest new parameters that can be taken into consideration when analysing student-performance and describing their cognitive state as well as parameters for describing exercises' characteristics.

We show that neural network approaches are worth considering as they overperform state-of-the-art models.

The objective for future works may be to introduce the task-level distraction sensitivity parameter (δ) as an explicit, learnable model parameter, which the model estimates automatically during the learning process based on student responses, rather than relying solely on expert-driven calibration. Another objective is for the model to infer the attentional state (φ) from the internal embeddings of the neural cognitive diagnosis model (CDM), through a dynamic analysis of learners' responses and estimated ability parameters, at the level of individual items and timepoints. This estimation is to be validated using quantitative error metrics to ensure reliability and accuracy.

8 Data availability

The datasets generated and/or analysed during the current study are available in the Zenodo repository, https://zenodo.org/records/18449944

9 References

[1] Barkley, R. A. (2006). *Excerpt from Attention-Deficit Hyperactivity Disorder: A Handbook for Diagnosis and Treatment* (2nd ed.). Guilford Press. Retrieved from https://www.guilford.com/excerpts/barkley2.pdf?t=1

[2] DiBello, L. V., Roussos, L. A., & Stout, W. (2006). A review of cognitively diagnostic assessment and a summary of psychometric models. *Handbook of Statistics*, *26*, 979–1030.

[3] Embretson, S. E., & Reise, S. P. (2013). *Item Response Theory*. New York: Psychology Press.

[4] Frederiksen, N., Mislevy, R. J., & Bejar, I. I. (1993). *Test Theory for a New Generation of Tests*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[5] Füstös, G. and Forstner, B. (2025). Data Synthesis and Simulation for Modeling Cognitive Abilities. Acta Polytechnica Hungarica, 22(6), pp. 119-135.

[6] Glaser, R. (1981). The future of testing: A research agenda for cognitive psychology and psychometrics. *American Psychologist*, *36*(9), 923–936.

[7] Hambleton, R. K., Swaminathan, H., & Rogers, D. J. (1991). *Fundamentals of Item Response Theory* (D. S. Fosler, Ed.). Thousand Oaks, CA: SAGE Publications.

[8] Liu, Q., Huang, Z., Chen, E., Zheng, Y., Chen, Y., Su, Y., & Hu, G. (2019). *Enhancing item response theory for cognitive diagnosis*. arXiv preprint arXiv:1905.10957.

[9] Lord, F. M. (1980). *Applications of Item Response Theory to Practical Testing Problems*. Hillsdale, NJ: Routledge.

[10] Nguyen, T.-N., Drumond, L., Krohn-Grimberghe, A., & Schmidt-Thieme, L. (2010). Recommender system for predicting student performance. *Procedia Computer Science*, *1*(2), 2811–2819.

[11] Pai, C.-W., Gopalakrishnan, V., & Beck, J. (2018). Estimating student ability and problem difficulty using item response theory and TrueSkill. *Information Discovery and Delivery*, *46*(4), 199–206.

[12] Sonuga-Barke, E. J. S. (2003). The dual pathway model of AD/HD: An elaboration of neuro-developmental characteristics. *Neuroscience & Biobehavioral Reviews*, *27*(7), 593–604. https://doi.org/10.1016/j.neubiorev.2003.08.005

[13] Sun, J., & Giles, C. L. (2005). An item response theory model for student ability evaluation using computer-automated test results. In *Advances in Web-Based Learning – ICWL 2005* (pp. 529–540). Springer.

[14] Tevesz, J., & Forstner, B. (2025). *Synthetic Data for Cognitive Diagnosis: Enabling Accurate and Efficient Inference with NeuroCDM.* In *Proceedings of the Automation and Applied Computer Science Workshop 2025 (AACS'25)*

[15] Wang, F., et al. (2024). A survey of models for cognitive diagnosis: New developments and future directions. July 2024.

[16] Wu, Z., Pan, Y., Ye, Q., & Kong, L. (2016). The City Intelligence Quotient (City IQ) evaluation system: Conception and evaluation. *Engineering*, *2*(2), 196–211.

10 Acknowledgements

This project was supported by the European Union project RRF-2.3.1-21-2022-00004 within the framework of the Artificial Intelligence National Laboratory.