Modelling Assessment Rubrics through Bayesian Networks: a Pragmatic Approach

Francesca Mangili *IDSIA (USI-SUPSI)*

Scuola Universitaria Professionale

della Svizzera Italiana (SUPSI)

Lugano, Switzerland

francesca.mangili@supsi.ch Giorgia Adorni *IDSIA (USI-SUPSI)*

Università della Svizzera Italiana (USI)

Lugano, Switzerland

giorgia.adorni@idsia.ch Alberto Piatti *DFA*

Scuola Universitaria Professionale

della Svizzera Italiana (SUPSI)

Locarno, Switzerland

alberto.piatti@supsi.ch Claudio Bonesana *IDSIA (USI-SUPSI)*

Scuola Universitaria Professionale

della Svizzera Italiana (SUPSI)

Lugano, Switzerland

claudio.bonesana@supsi.ch Alessandro Antonucci *IDSIA (USI-SUPSI)*

Scuola Universitaria Professionale

della Svizzera Italiana (SUPSI)

Lugano, Switzerland

alessandro.antonucci@supsi.ch

Abstract—**Automatic assessment of learner competencies is a fundamental task in intelligent tutoring systems. An assessment rubric typically and effectively describes relevant competencies and competence levels. This paper presents an approach to deriving a learner model directly from an assessment rubric defining some (partial) ordering of competence levels. The model is based on Bayesian networks and exploits logical gates with uncertainty (often referred to as noisy gates) to reduce the number of parameters of the model, so to simplify their elicitation by experts and allow real-time inference in intelligent tutoring systems. We illustrate how the approach can be applied to automatize the human assessment of an activity developed for testing**

computational thinking skills. The simple elicitation of the model starting from the assessment rubric opens up the possibility of quickly automating the assessment of several tasks, making them more easily exploitable in the context of adaptive assessment tools and intelligent tutoring systems.

Index Terms—Probabilistic reasoning, Noisy-OR Bayesian networks, Assessment rubrics, Computational thinking

I. INTRODUCTION

Intelligent tutoring systems (ITSs) are technological devices that support learning by interacting with the user, without the mediation of a teacher, supplying hints and suggestions which can be effective only if calibrated to the actual user competence level. Therefore, ITSs collect data during the accomplishment of the assigned tasks, analyse the learner activities and infer its competence profile based on a predefined model of the learner skills, knowledge and behaviours, and use it to select the most appropriate intervention. The new knowledge collected along with the learning activity continuously updates the competence profile, making the interventions more focused. Therefore, a central element in developing a successful AI-based educational tool is the learner model, aiming to describe the learner by a set of hidden variables representing competencies and their relations to the observable actions performed while solving the task.

A combination of knowledge, skills, and attitudes expressed in a context defines competencies. To assess them, teachers should set up realistic, authentic situations where students can apply their knowledge, skills and attitudes and compare the level of competence activated during these activities with a competence model, often specified through an assessment rubric. [1]. A general assessment rubric consists of a list of competence components to be assessed and a qualitative description of pos-

sible observable behaviours corresponding to different levels of such components. A rubric, therefore, describes the relationship between competencies and observable behaviours that one needs to codify in the learner model formally. Several sources of uncertainty and variability affect the relation between the non-observable competencies and the corresponding observable actions. A deterministic relation cannot correctly model it. Instead, probabilistic reasoning is a more appropriate approach to translating qualitative assessment rubrics into a quantitative, standardised, coherent measure of student proficiency. In the literature, *Bayesian knowledge tracing* (BKT), *Item response theory* (IRT), and *Bayesian networks* (BN) are all popular probabilistic approaches to model learner knowledge. BNs are a powerful framework for describing dependencies between skills and students' behaviours in facing complex tasks; furthermore, being graphical models highly intelligible, they are usable by experts in the elicitation of the student model. In Desmarais review of all most successful ITS experiences since Bloom seminal paper [2], it is acknowledged that "Probabilistic models for skill assessment are playing a key role in these advanced learning environments" and BNs are presented as the most general approach for modelling learner skills. Building on these results, we focused on BNs approaches.

Designing a BN may require a deep understanding of BNs theory and a significant effort in eliciting the network structure and parameters or the availability of a large dataset for learning the model directly from the data. Although BN arcs can be interpreted as a causal model, their definition by experts is not always trivial due to the complexity of causal relations at play and the presence of hidden causes. On the other hand, even when the learner model structure can be accurately defined, the elicitation or learning of the BNs parameters and the computation of inferences can

quickly become unmanageable. The number of parameters and the problem complexity can quickly increase with the number of arcs in the network. This issue can discourage ITS practitioners from using these tools in their applications when many skills are affecting the learner's actions. To avoid it, a solution to reduce the number of parameters in a BN-based learner model was proposed in our previous paper [3], which exploited the so-called *noisy-OR gates* [4]. We could reduce the number of parameters to elicit from exponential to linear for the number of parents/skills. Similar advantages also concern the inference. We adopted such solution to set up a general approach for translating assessment rubrics into interpretable BN-based learner models with a complexity compatible with real-time assessment. To illustrate this approach, we focused on the activity proposed in [5] for the standardised assessment of algorithmic skills along the entire K-12 school path, derived two learner models with different sets of expert-elicited parameters and applied them to the dataset collected in [3]. Overall, we obtain a compact and general approach to implementing a learner model given a set of competencies of interest and the corresponding assessment rubric. The resulting model has a simple structure and interpretable parameters, requiring a reasonable effort for their elicitation by experts and fast inferences allowing for real-time ITS interactions.

The paper is organised as follows: Section II provides some background about learner modelling based on BNs and noisy-OR gates; the approach is applied to a general assessment rubric in Section III; in Section IV we illustrate the procedure on the assessment rubric developed for the CAT activity, and analyse the model inferences based on the results of the pupils observed in the CAT study [5].

II. Exploiting Noisy Gates in BN-based Learner Models

A. BN-based learner model

The structure of a Bayesian network (BNs) over a set of variables is described by a directed acyclic graph $\mathcal{G}$ whose nodes are in one-to-one correspondence with the variables in the set. We call parents of a variable X, according to $\mathcal{G}$, all the variables connected directly with X with an arc pointing to it. Learner models usually include a set of n latent (i.e., hidden) variables $\boldsymbol{X} := (X_1, \ldots, X_n)$ (to be called *skill nodes* in the following) describing the competence profile of the learner and some m manifest variables $\boldsymbol{Y} := (Y_1, \ldots, Y_m)$ (answer nodes) describing the observable actions implemented by the learner to answer each specific task. Typically a bipartite structure with arcs from the skill nodes to the answer nodes, but not vice versa, is adopted. This structure is well-suited to model assessment rubrics, resulting in a simple and interpretable set of relations modelling the fact that the presence or absence of a competence directly affects the learner's answers to questions requiring such competence. For this work, we focus on the case of binary skill nodes, taking value 1 if the pupil possesses the skill, and binary answers nodes, indicating whether the pupil has shown the desired behaviour or not.

The example shown in Fig. 1 graphically depicts these relations. Long multiplication skill (X_2) can be applied to answer both multiplications, and therefore X_2 is a parent node for both answer nodes Y_1 and Y_2; instead, the multiplication in Y_2 cannot be computed by fingers, and thus there is no direct arc from X_1 to Y_2.

Once graph $\mathcal{G}$ structuring the BN is established, the definition of the BN over the $n + m$ variables $\boldsymbol{V} = (V_1, V_2, \ldots, V_{n+m})$ of the network, including both skill ($\boldsymbol{X}$) and answer ($\boldsymbol{Y}$), consists in a collection of condi-
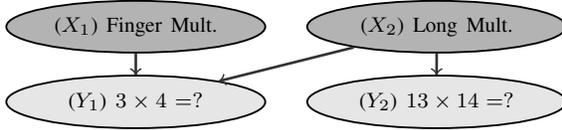
Fig. 1. Example of BN-based learner model. Adapted from [3].

tional probability tables (CPTs) giving the probabilities $P(Y_i = 1|\text{Pa}(Y_i))$ that Y_i takes value 1 given all possible joint states of its parent nodes $\text{Pa}(Y_i)$. Let $\boldsymbol{V}$ take values in $\Omega_{\boldsymbol{V}}$, the independence relations imposed from $\mathcal{G}$ by the *Markov condition* induce a joint probability mass function over the BN variables that factorises as follows:

$$P(\boldsymbol{v} = (\boldsymbol{x}, \boldsymbol{y})) = \prod_{v \in \boldsymbol{v}} P(v|\text{pa}(V)), \quad (1)$$

where $(v_1, v_2, \ldots, v_{n+m})$ represents a given joint state of the variable nodes in $\boldsymbol{V}$. BN inference consists in the computation of queries based on Eq. (1). In particular, we are interested in *updating* tasks consisting in the computation of the (posterior) probability mass function for a single skill node $X_q \in \boldsymbol{X}$ given the observed state y_E of the answer nodes $\boldsymbol{Y}_E \subseteq \boldsymbol{Y}$:

$$P(x_q|\boldsymbol{y}_E) = \frac{\sum_{\boldsymbol{v} \in \Omega_{\boldsymbol{V}'}} \prod_{v \in \boldsymbol{v}} P(v|\text{pa}(V))}{\sum_{\boldsymbol{v} \in \{\Omega_{\boldsymbol{V}'}, X_q\}} \prod_{v \in \boldsymbol{v}} P(v|\text{pa}(V))}, \quad (2)$$

where $\boldsymbol{V}' := \boldsymbol{V} \setminus \{\boldsymbol{Y}_E, X_q\}$.

According to the above model, multiple parent skills may be relevant to the same answer. If the answer node Y_j has n parent skills, this results in 2^n parameters to be elicited by experts. Besides the elicitation effort, also the inferential complexity can become critical. [3] discusses this and demonstrates how the use of noisy gates can avoids these issues. In the following section, we focus on the disjunctive noisy-OR gate, which shapes interchangeable skills and is suitable for modelling the assessment rubric of the activity from [5]. The proposed approach shares with BKT some similarities which will be emphasised later in this section, but, while the latter, in its common implementation, traces the evolution of a

single skill over time, the former focuses on fine-grained skills modelling in a specific moment.

B. Noisy-OR

The CPT of a *noisy-OR gate* is specified as [4]:

$$P(Y_j = 0|x_1, \ldots, x_n) = \prod_{i=1}^{n} (\mathbb{I}_{x_i=0} + \lambda_i \mathbb{I}_{x_i=1}), \quad (3)$$

where $\lambda_{ij} > 0$, $i = 1, \ldots, n$ are the model parameters defining the relation between Y_j and its n parent nodes, and $\mathbb{I}_A$ is an indicator function returning 1 if A is true and 0 otherwise. To better understand Eq. (3), a typical representation of the noisy-OR networks structure, introducing n auxiliary variables (also called inhibitor nodes), is shown in Fig 2. The state of Y_j is deterministically imposed as the logical disjunction (OR) of the auxiliary parent nodes. This first simplification removes the need of specify the answer node CPT given the state of its parent nodes. Furthermore, we set the input variable X_i as the unique parent of $X'_{i,j}$ and constrain $X'_{i,j}$ to be 0 with probability 1 when $X_i = 0$. Thus, the only parameter to be determined is $\lambda_{i,j} = P(X'_{i,j} = 0|X_i = 1)$. We can regard the auxiliary variable $X'_{i,j}$ as an *inhibitor* of skill X_i in performing the action described by Y_j, since with probability $\lambda_{i,j}$ it makes the skill unavailable to the success of Y_j even if the skill X_i is indeed mastered by the learner. It can be regarded as the analogous of the slip probability in BKT models.
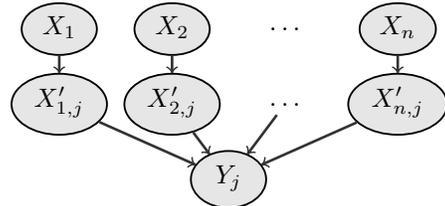

Fig. 2. A noisy gate (explicit formulation).

In accordance with the above description of the noisy-OR gate, missing skill i implies the inability to apply it

to any question j, whereas if the learner has the skill, the probability of being able to apply it depends on the specific task and is equal to $1 - \lambda_{i,j}$; the parameters of the model should therefore be related in some sense to the difficulty of the task. For instance, setting $\lambda_{i,j} = 1$, implies that having skill X_i has no effects on the capability of the learner to succeed in the task Y_j and that the inability to answer the question Y_j does not provide any information about the learner possessing skill X_i. This corresponds to a missing arc in the BN graph. On the other hand, $\lambda_{i,j} = 0$ means that the presence of skill X_i ensures that the learner will succeed in the task Y_j. Consequently, a question of this kind would be the most informative about skill X_i, especially in case of failure, since it would imply with probability 1 that the learner does not master skill X_i.

The noisy-OR can be used to describe a situation where a single skill is sufficient to answer a specific question, as in the case of Fig. 1 where multiplications are solved either by fingers or by long multiplication. In that example, the relations described by that network can be modelled by a noisy-OR with four parameters, one for each skill-answer pair, with $\lambda_{1,2} = 1$ to describe the fact that skill X_1 does not allow answering question Y_2.

To apply the above model, the domain expert (e.g., the teacher) should first list the parentless skill nodes $X_1, \ldots, X_n$ (with $X_i = 1$ meaning that the learner possesses skill i), the childless answer nodes $Y_1, \ldots, Y_m$ (with $Y_j = 1$ meaning a correct answer) and the parents skills relevant to each of them (setting to 1 the parameter λ_{ij} for all non-relevant skills). Then, he should quantify for each skill relevant to Y_j the value of $\lambda_{i,j}$ for a total of $n \cdot m$ parameters at most to be elicited. Finally, he should state the marginal prior probabilities π_i of each skill, which plays the role of the initial probabilities of possessing a skill in BKT. Notice that, since the

proposed approach does not model the learning process (differently from BKT which describes it through the transit probability), the concept of initial probability here represents our *initial* knowledge of the learner competence profile rather than its initial level.

Leaky Models: In a noisy-OR gate, when all skills are missing, all auxiliary variables are in the false state and, therefore, all answers must be wrong. Such model excludes the possibility of a lucky guess. To avoid this, the noisy-gates are made *leaky* by adding the leak node, a Boolean variable playing the role of an auxiliary skill $X_{j,leak}$, parent of Y_j and set to $X_{leak} = 1$. Parameter $1 - \lambda_{j,leak}$ describes the chances of guessing answer Y_j at random, i.e., without mastering any of the relevant competences. For instance, in a multiple choice question with four options, one of which is correct, one should set $1 - \lambda_{leak} = 1/4$. $1 - \lambda_{leak}$ can therefore be seen as the analogous of the guess probability in BKT.

Posterior Probabilities: After gathering the answers from the learner, the model computes posterior inferences about the probability of the learner possing each skill. When the given answer is wrong, i.e., Y_E is false, the noisy-OR implies that all its parent nodes $(X'_1, \ldots, X'_n)$ are in the false state, meaning that the learner was unable to apply any of the skill that would have led him to perform the desired action, either because the skills are indeed missing or because they were inhibited. Then, the posterior probability of having X_q when failing answer Y_E is related to the parameter $\lambda_{q,E}$ by

$$P(X_q = 1 | Y_E = 0) = \frac{\pi_q \lambda_{q,E}}{\pi_q \lambda_{q,E} + (1 - \pi_q)}, \quad (4)$$

implying a reduction in the probability that the student has the competence, the smaller the greater the inhibition probability $\lambda_{q,E}$. When instead the answer is correct, i.e., $Y_E = 1$, it is not possible to directly propagate the evidence to the auxiliary skill nodes since it can only be

stated that the learner was able to apply at least one of the competences relevant to Y_E. In this case, $P(X_q = 1|Y_E = 1)$ depends also on the parameters $\lambda_{j,E}$ assigned to the other parent skills X_j as follows:

$$P(X_q = 1|Y_E = 1) =$$
$$= \frac{\pi_q - \pi_q \lambda_{q,E} \prod_{j \neq q}(1 - \pi_j(1 - \lambda_{j,E}))}{1 - \prod_{j=1}^{n}(1 - \pi_j(1 - \lambda_{j,E}))}. \quad (5)$$

This implies an increased probability that the student has the competence, which is smaller the greater $\lambda_{q,E}$, since a skill with large inhibition probability is not likely to be the one which enabled the success in the task.

III. TRANSLATING ASSESSMENT RUBRIC INTO BNs

In this paper, we consider only assessment methods based on a task-specific assessment rubric [6] defined for assessing a given competence through a given task or family of similar tasks. IT consists of a two-entry table with rows corresponding to a component of the given competence, described in the light of the given task, and columns describing the competence levels, from initial to complete mastery of the competence component in the given task. During the solution of the task, for each combination of component and level, a qualitative description of the behaviour expected from a person with the given level in the given component is defined. Assessing a person's skill through a task-specific assessment rubric consists of matching the behaviours of the person solving the given task (or a battery of similar tasks) with those described in the assessment rubric to identify his/her competence level in each competence component. Each cell in the rubric is called competence level.

We define an ordering between competence levels by considering a competence level higher than another if the former is sufficient to perform all tasks requiring the latter. Each column of an assessment rubric represents the competence levels in increasing order. Therefore, all the competence levels are higher than those on their left. As shown in the CAT example below, when the competence components are conceptually overlapping, a hierarchical ordering between the rubric rows is also possible.

The competence level matching the student's behaviours may not match his/her actual level, e.g., if he is under-performing. When the task is composed of similar sub-tasks sharing the same assessment rubric, it is, therefore, possible to observe behaviours matching different competence levels in the different sub-tasks. This uncertainty is handled using the BN-based approach described in Section II to describe learner behaviours according to the defined assessment rubric probabilistically. Consider an assessment rubric with R rows and C columns. We define $R \cdot C$ hidden competence variables X_{rc} taking value 1 if the student masters the competence level corresponding to the assessment rubric's cell (r, c). For each task t (in a battery of T similar tasks) and each competence variable, we define an observable binary variable Y_{rc}^{t}, taking value 1 if the behaviour described in the cell (r, c) is observed while solving task t.

Variables X_{rc} and Y_{rc}^{t} represent, respectively, the skill and answer nodes of the network described in Section II. To account for the partial ordering defined by the assessment rubric, we set as parent nodes of each answer node Y_{rc}^{t} the skill node X_{rc} corresponding to the same cell of the assessment rubric, together with all other skill nodes referring to higher competence levels. This structure assumes that an observed behaviour can be explained by the student mastering the corresponding competence level or a higher one.

IV. A CASE STUDY ON K-12 ALGORITHMIC SKILLS

To illustrate our method, we consider the Cross Array Task (CAT), an unplugged activity to assess the algorithmic skills of pupils between 3 and 16 years of age
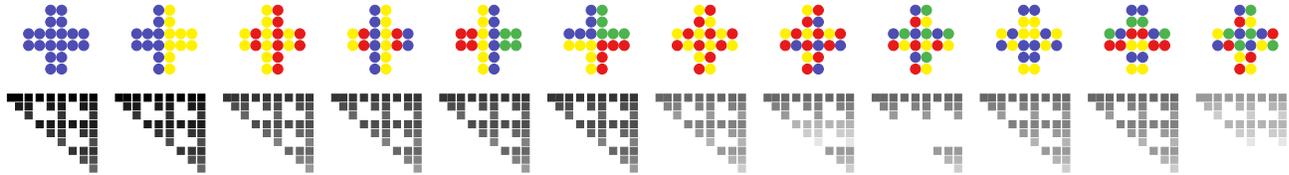
Fig. 3. Cross array schemes (top) and corresponding parameters (bottom). The rows represent answers, columns the skills, and darker shades of grey lower skill-answer inhibition probabilities (white for non-relevant skills).

[5]. Pupils are given a coloured cross array (Figure 3, top) and asked to provide the teacher instructions to reproduce the same colouring pattern. Different levels of complexity characterise the schemes. If challenged, pupils have two aids at their disposal: an empty CAT scheme (S) at which they can point to illustrate their instruction through gestures, and feedback (F), i.e., they can see how the other person is colouring the scheme. The instructions provided by each pupil, called an algorithm, are classified into three hierarchical categories. (1) *0D*: 0-dimensional algorithms are based on the use of Color-One-Dot (COD) operations only. (2) *1D*, structures such as rows, diagonal, squares etc. are also used. (3) *2D*, repetitions (loops) on dots or structures are also used. The complexity of the produced algorithms defines the competence components of the assessment rubric. The tools used to accomplish the task determine the rubric's levels (columns). The maximum level (V) is achieved when providing instructions by voice only, without seeing the scheme being coloured; level (VS) when requesting the help of the empty scheme; level (VSF) when also asking for the feedback. We have, thus, defined a CAT assessment rubric with three rows and three columns. Furthermore, a CAT score, ranging from 0 to 4 (Table I), was defined to summarise the performance of a pupil in a single scheme [5].

Besides ordering the levels in the columns of the rubric, we define an ordering also on the rows, since mastering algorithms of higher complexity implies mastering

TABLE I
DEFINITION OF THE CAT-SCORE METRIC.

	VSF	VS	V
0D	0	1	2
1D	1	2	3
2D	2	3	4

simpler ones. Therefore, we can say that competence levels X_{rc} are higher than $X_{r'c'}$ whenever $c > c'$ and $r >= r'$, or $c = c'$ and $r > r'$. When, instead, $c > c'$ but $r < r'$, neither skill can be said to dominate the other.

As described in the previous section, all competence levels in the rubric are assigned both a hidden variable X_{rc} (skill nodes) and an observable variable Y_{rc}^t (answer nodes) for each task $t = 1, \ldots, 12$. Observing $Y_{rc}^t = 1$ means that the pupil has solved the CAT t using an algorithm of complexity corresponding to the c-th row of the rubric, and asking the help admitted by the r-th column. As an example, $Y_{11}^k = 1$ means that the pupil has solved the k-th scheme by a 0D algorithm, using all helps (VSF). Theoretically, all answer nodes should be observed (or observable) through specific interactions with the learner. However, the dataset in [5], here used to test the proposed approach, was collected by proposing CATs to pupils and letting them choose the algorithm and the help they wished to use. We, therefore, encoded the collected answers as follows, to make such a dataset compatible with our model. A task t solved at level c^*

by an algorithm with complexity r^* was translated into $Y_{rc}^t = 1$ for all competence levels rc lower than or equal to r^*c^*, thus assuming that the pupil would have been able (if requested) to implement solutions requiring a lower competence level; similarly, we set $Y_{rc}^t = 0$ for all higher levels while leaving non-comparable answer nodes unobserved.

Concerning parameters' elicitation, uniform prior probabilities, i.e., $\pi_{rc} = 0.5$, are assigned to each skill, while two sets of values are considered for the inhibition probabilities $\lambda_{i,j}^t$. The first one, hereafter referred to as Model 1, is very basic, as it assigns the same value, namely $\lambda_{i,j}^t = 0.2$, to all parameters, except those corresponding to skills non-relevant to answer $Y_{i,j}^t$, in which case $\lambda_{i,j}^t = 1$. The goal of this model is to analyse the effect of the constraints resulting from the ordering of the skills alone. The second, called Model 2, aims at describing more in detail the difficulty that the pupils can encounter in applying their skills to different schemes. Left aside non-relevant skills for which $\lambda_{i,j}^t$ remains equal to 1, the strength of the remaining skill-question relation was given ten possible levels, namely, $\lambda_{i,j}^t \in 1 - \{0.45, 0.50, 0.55, 0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90\}$ depending on the characteristics of each CAT scheme. Our succinct elicitation setup allows summarizing graphically both the BN topology and its parameter values at the bottom of Figure 3. For both models, a leak node with guess probability $1 - \lambda_{\text{leak}} = 0.1$ has been associated to all answer nodes. We implemented the underlying BN within the CREMA Java library [7].

V. RESULTS

We considered the responses provided by all 109 pupils included in the study by Piatti et al. [5], calculated the student's average CAT scores over the twelve administered CAT schemes and compared them to probabilistic CAT scores computed for the BN-based models and defined as the expected number of competence levels mastered by the student, which corresponds to the sum of the marginal posterior probabilities of all skill nodes. The correlation between the original CAT score and the probabilistic scores is high (Pearson correlation coefficient of 0.94 for both BN-models) showing the consistency between the probabilistic assessment and the one by experts. The BN-based models, however, provide more detailed information about student competence profiles in the form of posterior probabilities for each competence level.

The probabilistic scores of the two BN-based models defined are similar. However, it is possible to discern relevant differences in the posterior probabilities of the individual skill nodes. To show this and to demonstrate the interpretability of the model results, we show in Table II hereafter the answers provided by three representative pupils and the corresponding posterior probabilities inferred by the models. Less straightforward is the situation of pupil 1 which was unsuccessful in using 1D algorithms with voice (1D-V) in schemes 3, 8, 10, 11, and 12 but was successful in all other schemes. Then, according to Model 1, which weights all schemes equally, he is not proficient in the 1D-V skill but only in the 1D-VS one (i.e., e must be supported by the empty scheme to produce a successfull 1D algorithm). Model 2 grants a larger probability to the 1D-V skill since it assigns larger inhibition probability to the 1D-V skill nodes when the task is more difficult. Therefore, failing to apply the 1D-V strategy to more difficult scheme does not reflect a lack of this skill if it is, instead, correctly implemented in easier tasks. Pupil 75 usually achieves medium to low-level solutions using 1D or 0D VS skills. He never approaches the problem using higher-level VS skills. Since 1D-VS fails on easy schemes, both models assign non-negligible probabilities to 0D-VS and

TABLE II

POSTERIOR PROBABILITIES $P(X_{rc} = 1|\boldsymbol{y}^{(j)})$ OF MODEL 1/MODEL 2 FOR THREE REPRESENTATIVE PUPILS.

Student	$P(X_{rc} = 1\|\boldsymbol{y}^{(j)})$								
	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}
j	0D-VSF	0D-VS	0D-V	1D-VSF	1D-VS	1D-V	2D-VSF	2D-VS	2D-V
1	0.54/0.55	0.70/0.76	0.92/0.98	0.64/0.66	0.71/0.89	0.17/0.99	0.71/0.74	0.21/0.79	0.00/0.05
21	0.54/0.55	0.70/0.76	0.99/1.00	0.69/0.74	0.97/0.99	1.00/1.00	0.89/0.93	1.00/1.00	1.00/1.00
75	0.54/0.55	0.31/0.50	0.00/0.01	0.64/0.66	0.00/0.04	0.00/0.00	0.07/0.19	0.00/0.00	0.00/0.00

1D-VSF. Notice that, since low competence levels are associated with fewer answer nodes, their presence is less accurately assessed by the administered tasks than that of higher levels.

VI. CONCLUSIONS

In this work, we have proposed a procedure for deriving a learner model for automatic skill assessment directly from the competence rubric of any set of tasks. The approach has been illustrated and its feasibility demonstrated by implementing the automatic assessment of the cross-array task [5]. Results show that the derived model and its inferences can be easily interpreted. The model will be used in an application for the adaptive assessment of pupil computational skills.

The merits of the approach are the simple elicitation, and the fast inferences resulting from the use of nosy-OR BNs [3]. Its limitations are mainly two: either disjunctive or conjunctive gates must be used, while sometimes it would be useful to combine both or implement more general relations between skills; although partially implied by the structure of the model, the ordering between competence levels is not strictly enforced. Therefore, the current work extends the model to overcome these limitations while avoiding increasing the computational burden significantly.

REFERENCES

[1] P. Dawson, "Assessment rubrics: towards clearer and more replicable design, research and practice," *Assess Eval High Educ*, vol. 42, no. 3, pp. 347–360, 2017.

[2] M. C. Desmarais and R. S. d Baker, "A review of recent advances in learner and skill modeling in intelligent learning environments," *User Model User-adapt Interact*, vol. 22(1), pp. 9–38, 2012.

[3] A. Antonucci, F. Mangili, C. Bonesana, and G. Adorni, "Intelligent tutoring systems by bayesian networks with noisy gates," in *35th FLAIRS Conference*. Florida Online Journals, 2022.

[4] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[5] A. Piatti, G. Adorni, L. El-Hamamsy, L. Negrini, D. Assaf, L. Gambardella, and F. Mondada, "The CT-cube: A framework for the design and the assessment of computational thinking activities," *Comput. Hum. Behav. reports*, vol. 5, pp. 100–166, 2022.

[6] A. Jonsson and G. Svingby, "The use of scoring rubrics: Reliability, validity and educational consequences," *Educ. Res. Rev.*, vol. 2(2), pp. 130–144, 2007.

[7] D. Huber, R. Cabañas, A. Antonucci, and M. Zaffalon, "Crema: A java library for credal network inference," in *10th PGM Conference*, 2020, pp. 613–616.