Enhancing the Error Diagnosis Capability for Constraint-Based Tutoring Systems

Nguyen-Thinh Le and Niels Pinkwart

Clausthal University of Technology, Germany
{nguyen-thinh.le,niels.pinkwart}@tu-clausthal.de

Abstract. Constraint-based modelling techniques have been demonstrated a useful means to develop intelligent tutoring systems in several domains. However, when applying CBM to tasks which require students to explore a large solution space, this approach encounters its limitation: it is not well suited to hypothesize the solution variant intended by the student, and thus corrective feedback might be not in accordance with the student's intention. To solve this problem, we propose to adopt a probabilistic approach for solving constraint satisfaction problems.

Keywords: ITS, weighted constraint-based model, cognitive diagnosis.

1 Introduction

The constraint-based modelling (CBM) approach [5] has been successfully employed in several domains, such as diagnosing grammar errors in natural languages [3], building intelligent tutoring systems for SQL [4]. One of the strengths of this approach is that it does not require an enumeration of every correct solution for modelling, nor is it necessary to anticipate possible errors made by students. Instead, a number of domain principles and properties of correct solutions for a problem need to be specified. However, this approach encounters its limitation when applying it to tasks which have a large solution space. Corrective feedback derived from results of constraint-based error diagnosis, might be misleading, because the solution strategy the student intended to implement is not the same one the constraints are based on. This problem has been identified and discussed in [2] and [6]. This problem raises the need to hypothesize the student's intention in terms of the applied solution strategy during the process of diagnosing errors. Once the solution strategy of the student has been identified, it makes sense to evaluate constraints in the context of that specific solution strategy only. This paper introduces a weighted constraint-based model adopting a probabilistic approach for solving constraint satisfaction problems: each constraint is enriched with a weight value indicating the importance of the constraint. Applying this model, a tutoring system is able to decide on the most plausible hypothesis about the solution strategy intended by the student.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 500–503, 2011.

2 A Weighted Constraint-Based Model For ITS

In order to be able to identify shortcomings in a student solution and to provide appropriate corrective hints according to the solution strategy pursued by the student, a tutoring system needs to cover a space of possible solutions and the student solution needs to be analyzed thoroughly. In the approach proposed in this paper, the weighted constraint-based model serves these two purposes.

Semantic Table: Instead of using a single ideal solution to capture problem-specific requirements as in [2], the model introduced in this paper uses a so-called *semantic table* which comprises two ideas: 1) it models several solution strategies, and 2) it represents model solutions in a relational form. The first characteristic serves to hypothesize the most plausible strategy underlying a student solution. The second one has the advantage that solution variants (e.g., created by alternative orderings of solution components) can easily be covered. The following table illustrates a partial semantic table for the problem *"Calculate the return after investing an amount of money at a constant yearly interest rate"*, covering one possible solution strategy (tail recursive), where *CI* and *SI* are abbreviations for clause index and subgoal index, respectively.

Strategy	CI	Head	SI	Subgoal	Description
Tail recursive	1	p(S,_,P,Ret)	1	P=0	Recursion stops
	1	p(S,_,P,Ret)	2	Ret=S	Recursion stops
Tail recursive	2	p(S,R,P,Ret)	1	P>0	Check period
	2	p(S,R,P,Ret)	2	NS is S*R+S	Calculate new sum
	2	p(S,R,P,Ret)	3	NP is P-1	Update period
	2	p(S,R,P,Ret)	4	p(NS,R,NP,Ret)	Recur with new period

Constraints: We distinguish between *general constraints* and *semantic constraints*. Constraints of the former type are used to model domain-specific principles, which every solution variant of any problem must adhere to and are independent of problem-specific requirements. For instance, in programming, to evaluate an arithmetic expression, all variables must be instantiated. Such a domain principle can be modeled by means of general constraints which can be instantiated by the following constraint schema, where the problem situation X and the condition Y can be composed of elementary propositions using conjunction or disjunction operators.

IF problem situation X is relevant **THEN** condition Y must be satisfied

Semantic constraints are used to check the semantic correctness of a student solution. Constraints of this type require problem-specific information specified in the semantic table and have the following schema, where *STS* is an abbreviation for *student solution*.

IF in the semantic table, a component X exists and satisfies condition α
THEN in the STS, a corresponding component exists and satisfies α

Transformation Rules: To extend the solution space for a problem that involves mathematical expressions, transformation rules can be defined based on mathematical theorems, e.g., distributive and commutative laws.

Constraint Weights: As pointed in Section 1, constraint-based tutoring systems might provide misleading corrective feedback. We need a means to search the most plausible hypothesis about the student's solution variant. For this purpose, we exploit approaches to softening constraints in constraint satisfaction problems (CSP). The weighted constraint-based model proposed here adopts the probabilistic CSP approach [1] for error diagnosis. Following this approach, each constraint is attached a *constraint weight*, indicating the measure of importance. Weight values are taken from the interval $[0;1]$. The value close to 0 indicates the weight for constraints which model most important requirements. Constraints of the latter type can be considered *hard constraints*. The plausibility of each hypothesis is calculated using the formula: $Plausibility(H) = \prod_{i=1}^{N} W_i$, where W_i is the weight of a violated constraint.

Error Diagnosis: Given a student solution, the process of error diagnosis starts to match the solution against each of the solution strategies specified in the semantic table. This process initialises *global mappings* representing hypotheses about the strategy underlying the student solution and this level of matching is referred to as *strategy level*. Then, the process continues to generate hypotheses about the student's solution variant by matching the components of the student solution against the corresponding ones of the selected solution strategy. The matching process results in *local mappings* representing hypotheses about the student's solution variant. They are used to complete global mappings. This level of matching is called *solution variant level*. After hypotheses have been generated, the process of error diagnosis evaluates each hypothesis with respect to its plausibility. On the solution variant level, the most plausible solution variant of the student solution is determined by choosing the hypothesis with maximal plausibility score. On the strategy level, the hypothesis with the highest plausibility score is considered the solution strategy being implemented in the student solution. Diagnostic information is derived from constraint violations based on the best hypothesis.

3 Conclusion

In this paper, we have argued that the classical CBM approach is not well-suited to build intelligent tutoring systems for tasks which have a large solution space. Here, the process of error diagnosis needs to hypothesize the solution variant applied by the student. For this purpose, we introduced the weighted constraint-based model which adopts the idea of probabilistic techniques for solving constraint satisfaction problems.

References

1. Fargier, H., Lang, J.: Uncertainty in Constraint Satisfaction Problems: a Probabilistic Approach. In: Moral, S., Kruse, R., Clarke, E. (eds.) ECSQARU 1993. LNCS, vol. 747, pp. 97–104. Springer, Heidelberg (1993)
2. Martin, B.: Intelligent Tutoring Systems: The Practical Implementation Of Constraint-based Modelling. PhD thesis, University of Canterbury (2001)

3. Menzel, W.: Diagnosing Grammatical Faults - a Deep-modelled Approach. In: AIMSA, pp. 319–326 (1988)
4. Mitrovic, A., et al.: Constraint-based Tutors: A Success Story. In: 14th Int. Conf. on Industrial and Engineering Appl. of AI and Expert Systems, pp. 931–940 (2001)
5. Ohlsson, S.: Constraint-based Student Modeling. In: Greer, J.E., et al. (eds.) Student Modelling: The Key to Individualized Knowledge-based Instruction, pp. 167–189. Springer, Heidelberg (1994)
6. Woolf, B.P.: Building Intelligent Interactive Tutors. Morgan Kaufmann, San Francisco (2009)