Human-Adaptive Determination of Natural Language Hints

Dimitra Tsovaltzi and Armin Fiedler
P.O. Box 15 11 50, D-66041 Saarbrücken, Germany
{tsovaltzi,afiedler}@ags.uni-sb.de

Abstract

Putting emphasis on making cognitive functions of hints explicit and allowing for the automatic incorporation in a natural dialogue context which simulates the dialectic process of a socratic teaching model, we developed a multi-dimensional hint taxonomy where each dimension defines a decision point for the associated function. We take domain knowledge into account for choosing the hint to be given with regard to the domain-knowledge dimension. We show how structured domain knowledge can be used to inform the automatic determination of the hint category to be produced and assists natural argument interaction in an ITS context.

1 Introduction

Empirical evidence has shown that natural language dialogue capabilities are a crucial factor to making human explanations effective [Moore, 1993]. Moreover, the use of teaching strategies is an important ingredient for intelligent tutoring systems. Such strategies, normally called dialectic or *socratic*, have been demonstrated to be superior to pure explanations, especially regarding their long-term effects [Chi *et al.*, 1994; Rosé *et al.*, 2001; Ashley *et al.*, 2002]. Consequently, an increasing though still limited number of state-of-the-art tutoring systems use natural-language interaction and automatic teaching strategies, including some notion of hints, which implement the socratic style of teaching. *Hints* are defined as the means for encouraging active learning. They take the form of eliciting information that the student is unable to access without the aid of prompts, or information which she can access but whose relevance she is unaware of with respect to the problem at hand. A hint can also point to an inference that the student is expected to make based on knowledge available to her [Hume *et al.*, 1996b].

CIRCSIM-Tutor [Hume *et al.*, 1996a], an intelligent tutoring system for blood circulation, applies a taxonomy of hints, relating them to constellations in a planning procedure that solves the given tutorial task. Moreover, it uses a domain ontology to categorise the student answer and to fix mistakes. AutoTutor [Person *et al.*, 2000] uses curriculum scripts on which the tutoring of computer literacy is based, where hints are associated with each script. AutoTutor also aims at making the student articulate expected answers and does not distinguish between the cognitive function and the dialogue move realisation of hints. The emphasis is on self-explanation, in the sense of re-articulation, rather than on trying to help the student to actively produce the content of the answer herself. Matsuda and VanLehn [2003] research hinting for helping students with solving geometry proof problems. They orient themselves towards tracking the student's mixed directionality, which is characteristic of novices, rather than assisting the student with specific reference to the directionality of a proof. Melis and Unllrich [2003] are looking into Polya scenarios in order to extract possible hints. They aim these hints for a proof presentation approach. Within the framework of STEVE [Rickel *et al.*, 2000], Diligent [Angros *et al.*, 2002] is a tool for learning domain procedural knowledge and providing respective explanations. Knowledge is acquired by observing an expert's performance of a task, as a first step, subsequently conducting self-experimentation, and finally by human corrections on what Diligent has taught itself. The relevant part to our work is the knowledge representation, and in that the representation of procedures in terms of steps in a task, ordering constraints, causal links and end goals. STEVE is currently limited in making use of the procedure representation to learning text in order to provide explanations.

On the whole, current models of hints are somehow limited in capturing their various underlying functions explicitly and relating them to the domain knowledge dynamically. Moreover, they do not distinguish dialogue model from teaching model considerations, making hinting non-flexible. Both of these aspects hinder the naturalness of the argumentation in the process of the tutoring task towards constructing the required proof.

Our approach is oriented towards integrating hinting in natural language dialogue systems, where the socratic teaching and its dialectic nature can be implemented [Tsovaltzi and Karagjosova, 2004]. We model a socratic teaching strategy, which allows us to manipulate aspects of learning, such as help the student build a deeper understanding of the domain, eliminate cognitive load, promote schema acquisition, and manipulate motivation levels [Wilson and Cole, 1996; Lim and Moore, 2002; Weiner, 1992], within natural language dialogue interaction. In contrast to most existing tutorial systems, we make use of the specialised domain reasoner

ΩMEGA [Siekmann *et al.*, 2002]. Moreover, we capture aspects of human reasoning by means of cognitive schemata, and by abstracting from the purely logical basis of the domain via a domain mathematical domain ontology adapted to tutoring needs [Tsovaltzi and Fiedler, 2003b].

Thus, putting emphasis on dynamically producing hints and making cognitive functions of hints explicit, we developed a multi-dimensional hint taxonomy where each dimension defines a decision point for the associated function [Tsovaltzi *et al.*, 2004]. The domain knowledge dimension is the way to structure and manipulate domain knowledge for tutoring decision purposes and generation considerations within a tutorial manager. We also use the domain knowledge to inform both the choice of an appropriate hint class and the specification of the actual hint to be produced. The discourse management aspects of the dialogue manager can be independently manipulated. Thus, the final sentence realisation of hints becomes dialogue adaptive and allows for the best of both worlds in tutoring dialogue systems: domain-knowledge-specific feedback and dialogue capabilities in tutoring.

Natural argumentation evolves in two fashions. First, in the dialectic of reaching a common ground towards resolving the proof task in the socratic teaching strategy which we model. The two parties in the argumentation process are (i) the student, free to insert any typed input while collaborating with the tutor in finding a proof, and (ii) the system, trying to reason with the student on the basis of her input towards a valid proof and aiming at the construction of *schemata* for the student, that is, an abstract re-applicable reasoning and argumentation line [Wilson and Cole, 1996]. Second, natural argument is introduced in keeping the mathematical logic of the formal proof as the basis for the hints produced, but capturing it in a human-oriented way for the evaluation of the student input and the hint realisation.

We originally based our work on data collected in the domain of basic electricity and electronics [Rosé *et al.*, 2001; Tsovaltzi, 2001], and defined a first mathematical ontology for our domain information needs [Tsovaltzi and Fiedler, 2003b]. We tested our approach to hinting in an experiment where 22 subjects were asked to prove simple theorems in our domain via dialogue interaction [Benzmüller *et al.*, 2003; Wolska *et al.*, 2004]. According to the analysis of the collected data, we made adaptations in all areas of our research. The augmented domain ontology and the proposed manipulations presented in this paper, are the results of these adaptations in the area of domain knowledge.

The structure of the paper is as follows: Section 2 looks at the way we model hint categories. Section 3 introduces briefly our domain ontology and describes its use for hint determination. Section 4 concludes the paper.

2 Hint Dimensions

On the one hand, the motivation for this work derives from the need to formalise the cognitive functions which underly hints, in order to produce adequate and psychologically justified feedback. On the other hand, we aim at separating out such underlying functions of hints from dialogue move functions, which might be common for different cognitive functions. This allows us to profoundly investigate both aspects and to take them into account sufficiently for the feedback. The actual sentence-level realisation of a hint will then be based on decisions regarding the function that better serves the tutoring goals, as well as decisions regarding the dialogue and discourse context, which take advantage of natural language dialogue capabilities.

To capture all the functions of a hint, which ultimately aim at eliciting the relevant inference step in a given situation, a hint category is described by the combination of four dimensions. In effect, the dimensions capture different aspects of the tutors argument. The ***domain knowledge*** dimension captures the needs of the domain, distinguishing different *anchoring points* for skill acquisition in problem solving, that is, central strategic points in the schema acquisition process, which can be used for instruction purposes and around which a schema may be built. The ***inferential role*** dimension captures the type of reasoning about the anchoring points. The ***elicitation status*** dimension distinguishes between information being elicited and degrees to which information is provided. The ***problem referential perspective*** dimension distinguishes between views on discovering an inference, including conceptual and pragmatic perspectives.

All combinations are potentially useful[1], even if it is for different teaching models. The combinations of hints modelled can be elaborations on anchoring points suitable for the domain, but also capture the educational values pertaining to the teaching model of choice. The abstract definition of the hint categories makes possible the free integration of other aspects of tutorial dialogues [Tsovaltzi *et al.*, 2004].

3 Manipulating the Domain for Adaptive Hinting

Having captured the domain knowledge in the respective hinting dimension, the role it plays dynamically is twofold. First, it influences the *choice* of the appropriate hint category by a socratic tutoring strategy [Fiedler and Tsovaltzi, 2003], and, second, it determines the *content* of the hint to be generated. Our general evaluation of the student input relevant to the task, the *domain contribution*, is defined based on the concept of expected proof steps, that is, valid *proof steps* according to some formal proof. In order to avoid imposing a particular proof solution and so to allow the student to follow her preferred line of reasoning, as the socratic model requires [Tsovaltzi and Fiedler, 2003a], we use the theorem prover ΩMEGA [Siekmann *et al.*, 2002] to test whether the student's contribution matches an expected proof step.

By comparing the domain contribution with the expected proof step we first obtain an overall assessment of the student input in terms of generic evaluation categories, such as correct, wrong, or near-miss. Second, we track abstractly defined domain knowledge that is useful for tutoring in general and applied in this domain. For this purpose, we defined a domain ontology [Fiedler and Tsovaltzi, 2003]. Both the domain contribution category and the domain ontology consti-

[1] For a hint category example see Section 3.2

tute a basis for the choice of the hint category that assists the student [Tsovaltzi *et al.*, 2004].

3.1 Ontology

Our ontology, which captures the domain needs for hinting, constitutes an enhancement on the existing domain ontology of the proof planner ΩMEGA, which is organised as a hierarchy of nested mathematical theories. Each theory includes definitions of mathematical concepts, lemmata and theorems about them, and inference rules, which can be seen as lemmata that the proof planner can directly apply. Moreover, each theory inherits all definitions, lemmata and theorems as well as all inference rules from nested theories. Since these draw on mathematical concepts defined in the mathematical theories, the mathematical database implicitly represents many relations that can be made use of in tutorial dialogue, in terms of identifying anchoring points that need to be addressed, as well as for capturing and making use of student reasoning, which is the basis for learning, but does not strictly abide to the rules of logic. In particular, we defined relations between mathematical concepts, anchoring points relating to the inference rules connecting two steps, relations between mathematical concepts and inference rules, and anchoring points for reasoning towards the inference for a step [Tsovaltzi and Fiedler, 2003b]. Examples for some new anchoring points are the *relevant concept* (intuitively, the main concept that is manipulated in a proof step and appears in the expression to be manipulated, the *source*) and the *subordinate concept* (intuitively, the second most relevant concept, which supports the relevant concept and appears in the resulting expression after the rule application, the *target*). These are dependent on the applicable inference rule.

3.2 Domain Knowledge for Hint Determination

The input to the socratic algorithm, which chooses the appropriate hint category to be produced, is given by the *hinting session status* (HSS), a collection of parameters that cover the student modelling necessary for our purposes. The HSS is only concerned with the current hinting session but not with inter-session modelling, and thus does not represent if the student recalls any domain knowledge between sessions.

The HSS can be further divided into two parts. The *global hinting session status* (GHSS) consists of a choice of counts relevant to the motivation levels of the student, such as the number of steps performed, the number of wrong answers in the session, and the number of times the student asks for additional help. These counts inform the decision to produce a hint or another dialogue move and, in the former case, the choice of the parameters in the hint dimensions. The specific hint category is then chosen based on the *local hinting session status* (LHSS), which represents the domain knowledge the student actually uses in the current session for the overall proof and in the particular step. Therefore, the LHSS represents additional information such as the domain contribution category and the number of wrong answers for the particular proof step.

In the LHSS, we represent those concepts or relations from the domain ontology, for which we found a pedagogically motivated use. Hence, we represent if the student has completed the proof step; if she has used or knows the relevant concept, the subordinate concept, a domain relation, the inference rule, the proof technique, the premise or the conclusion of the theorem to be proven, the proof direction, and if the proof is direct or indirect. Note that most of these concepts do not exist in the formal proof representation, and that none of them are mentioned to the student by their technical name, but only in a NL descriptive manner. For example, we use *inference rule* to mean the rule given in the assumed lesson material (in our domain, mathematical lemmata, theorems, etc.), which leads from the *source* to the *target* expression in every proof step. However, in the case of the inference rule *quantifier introduction*, for instance, the quantifier in a backward reasoning step, which is what the student is dealing with, is intuitively eliminated rather than introduced, and replaced by something that can be proven, but preserves the validity for all variables. Therefore, the rule can be referred to as "getting rid of the quantifier". Another case is *proof method*, an anchoring point which is pedagogically inadvisable to mention to the student directly, but is necessary for internal system communication protocols. For example, its instance, *Proof direction* captures whether the proof step is forward (i.e., form the assumptions towards the goal) or backward (i.e., from the goal towards the assumptions). According to mathematics didactics, such terminology used in formal proving has no place in tutoring proofs [Wu, 1996; 2001]. A hint dealing with this anchoring point is, hence, realised through reference to it. The student is pointed at applying rules to manipulate either the assumptions and their consequences (for forward steps), or to find expressions from which the goal can be inferred (for backward steps).

In addition to the domain knowledge used we also represent the previous hint category. Because hint categories deal with domain knowledge, we thus capture both the domain knowledge the student should know, as it has been previously given via a hint, as well as the kind of assistance already provided. Previous research shows that strong hints giving domain knowledge away increase the possibility of giving a correct answer, but do not necessarily mean that the domain knowledge is learned [Gertner *et al.*, 1998]. Hence, we check for the use of domain knowledge independently and record the number and kind of hints given and how much domain knowledge has been given away. Moreover, we use a *global hinting session aggregate*, that is a mean based on the GHSS, to assess the student performance in the session. The socratic algorithm gives progressively more informative hints and produces an *align* dialogue move when the student performance is low to check if the student understands the correct answer that she has given (cf. [Fiedler and Tsovaltzi, 2003] for details of the choice of hint categories).

Once the hint category has been chosen, the domain knowledge is used again to instantiate the category yielding a *hint specification*. Each hint category is defined based on generic descriptions of domain objects or relations, that is, the anchoring points. The role of the ontology is to assist the domain knowledge module (where the proof is represented) with the mapping of the generic descriptions on the actual objects or relations that are used in the particular context, that is, in the particular proof and the proof step under consideration. The relevant information will then be passed on to the generator. Different hint categories may make use of the same hint specifications.

Since the aim of our research is to avoid the standard static repertoire-based generation of hints, additional information for the final natural language realisation must be provided. Every hint category is chosen with respect to the HSS, hence the hint categories capture also all information encoded in the different dimensions. Additionally, discourse structure and dialogue modelling will bring into the realisation their own information. This means that for every student and for her current performance on the proof being attempted, a hint category is chosen, which must be realised in a different way based on the discourse structure as it deploys itself independently of the domain knowledge. The generator is the final place where all the information contributes to the sentence realisation (cf. [Tsovaltzi and Karagjosova, 2004]). This makes a flexible tutorial dialogue manager, which takes advantage of dialogue modelling, teaching models and natural language techniques for the complicated tutoring task.

To help the reader picture a hint category, but without making any claims as to the actual realisation of it, the hint *give-away-meta-reasoning-subordinate-concept* refers to the domain object subordinate concept at the domain knowledge dimension, it is meta-reasoning at the inferential role dimension, passive at the elicitation status dimension, and conceptual at the problem referential dimension. Assuming that the inference rule is *universal quantifier introduction*, and the subordinate concept what needs to be proved for the validity of the expression to hold universally, here is a possible realisation: "In order for the expression to hold for all x we need to prove it for an arbitrary constant a."

In other words, we represent the anchoring points and relations defined in our domain ontology, which are required for building both the proof and also mental schemata for constructing the proof. These anchoring points abstract from the strict mathematical logic of the formal proof and are adapted to the human oriented heuristics aiming at schema acquisition.

In terms of the cognitive and pedagogical motivation of our approach, we want to generate instructions to help the student learn based on some kind of heuristics [Wu, 1996]. We hint at the anchoring points to support creating the heuristics and we specify these anchoring points based on the structured domain ontology to give her the required content [Wu, 2001]. Moreover, we use additional meta-reasoning anchoring points to reinforce the main anchoring points, when necessary and appropriate [Lim and Moore, 2002], that is, when the student seems to need the reinforcement, but not if she is too weak to be introduced to new concepts. In the latter case we prefer giving more information at the performable-step level, leaving the meta-reasoning for a more appropriate moment in the student's learning. The schemata that the student builds from these hints are established on the anchoring points, but the exact way they are built and their structure depends on every student and cannot be tracked.

4 Conclusion

We have presented a way of automating hinting that allows for taking care of independent aspects of student feedback in a natural language tutorial dialogue system. We have explained how the manipulation of the domain knowledge can also be done in such a way, that it does not prohibit the natural language hint generation flexibility, but promotes it. The whole research presented here is inspired by the necessity to guide students through the task, while at the same time making use of everything they have at their disposal and can assist learning. Therefore, we let them build new knowledge on their own cognitive structures and only intervene when these are insufficient. To do so, we allow them to use their own human reasoning and abstract from the underlying logic of proving necessary for the system implementations. Such an architecture allows for the dialogue, discourse natural language generation and pedagogical capabilities of the system to be clearly defined and finally combined for the final realisation of the feedback. What remains to be implemented is the enhanced ontology build on top of the existing ontology, while the sentence realisation of the hints is still under research.

References

[Angros *et al.*, 2002] Richard Angros, Jr., W. Lewis Johnson, Jeff Rickel, and Schorel Andrew. Learning domain knowledge for teaching procedural skills. In *Proceedings of AAMAS'02*, Bologna, Italy, 2002.

[Ashley *et al.*, 2002] Kevin D. Ashley, Ravi Desai, and John M. Levine. Teaching case-based argumentation concepts using dialectic arguments vs. didactic explanations. In S. A. Cerri, G. Gouardéres, and F. Paraguaçu, editors, *Proceedings of the 6th International Conference on Intelligent Tutoring Systems*, pages 585 – 595. Berlin: Springer Verlag. A. L. Brown and M. J. Kane, 2002.

[Benzmüller *et al.*, 2003] Christoph Benzmüller, Armin Fiedler, Malte Gabsdil, Helmut Horacek, Ivana Kruijff-Korbayová, Manfred Pinkal, Jörg Siekmann, Dimitra Tsovaltzi, Bao Quoc Vo, and Magdalena Wolska. A wizard-of-oz experiment for tutorial dialogues in mathematics. In Vincent Aleven, Ulrich Hoppe, Judy Kay, Riichiro Mizoguchi, Helen Pain, Felisa Verdejo, and Kalina Yacef, editors, *AIED2003 — Supplementary Proceedings of the 11th International Conference on Artificial Intelligence in Education*, volume VIII: Advanced Technologies for Mathematics Education, pages 471 – 481, Sidney, Australia, 2003. School of Information Technologies, University of Sydney.

[Chi *et al.*, 1994] Michelene T. H. Chi, Nicholas de Leeuw, Mei-Hung Chiu, and Christian Lavancher. Eliciting self-explanation improves understanding. *Cognitive Science*, 18:439 – 477, 1994.

[Fiedler and Tsovaltzi, 2003] Armin Fiedler and Dimitra Tsovaltzi. Automating hinting in an intelligent tutorial system. In *Proceedings of the IJCAI Workshop on Knowledge Representation and Automated Reasoning for E-Learning Systems*, pages 23 – 35, Acapulco, 2003.

[Gertner *et al.*, 1998] Abigail S. Gertner, Christina Conati, and Kurt VanLehn. Procedural help in Andes: Generating hints using a bayesian network student model. In *Proceedings of the 15th National Conference on Artificial Intelligence*, pages 106 – 111, Madison, Wisconsin, 1998. AAAI Press/MIT Press.

[Hume *et al.*, 1996a] Gregory Hume, Joel Michael, Allen Rovick, and Martha Evens. Student responses and follow up tutorial tactics in an ITS. In *Proceedings of the 9th Florida Artificial Intelligence Research Symposium*, pages 168 – 172, Key West, FL, 1996.

[Hume *et al.*, 1996b] Gregory D. Hume, Joel A. Michael, Allen A. Rovick, and Martha W. Evens. Hinting as a tactic in one-on-one tutoring. *Journal of the Learning Sciences*, 5(1):23 – 47, 1996.

[Lim and Moore, 2002] Eng Leong Lim and Dennis W. Moore. Problem solving in geometry: Comparing the effects of non-goal specific instruction and conventional worked examples. *Journal of Educational Psychology*, 22(5):591 – 612, 2002.

[Matsuda and VanLehn, 2003] Noboru Matsuda and Kurt VanLehn. Modelling hinting strategies for geometry theorem proving. In P. Brusilovsky, A. Corbett, and F. de Rosis, editors, *Proceedings of the 9th International Conference on User Modeling*, pages 373 – 377, Pittsburgh, PA, 2003. Berlin, Heidelberg: Springer.

[Melis and Ullrich, 2003] Erica Melis and Carsten Ullrich. How to teach it — Polya-inspired scenarios in ActiveMath. In U. Hoppe, F. Verdejo, and J. Kay, editors, *Proceedings of AIED2003 Conference of Artificial Intelligence in Education*, pages 141 – 147, Biarritz, France, 2003. IOS Press.

[Moore, 1993] Johanna Moore. What makes human explanations effective? In *Proceedings of the Fifteenth Annual Meeting of the Cognitive Science Society*, Hillsdale, NJ, 1993. Lawrence Erlbaum Associates.

[Person *et al.*, 2000] Natalie K. Person, Arthur C. Graesser, Derek Harter, Eric Mathews, and the Tutoring Research Group. Dialog move generation and conversation management in AutoTutor. In Rosé and Freedman [2000], pages 45 – 51.

[Rickel *et al.*, 2000] Jeff Rickel, Rajaram Ganeshan, Charles Rich, Candance L. Sidner, and Neal Lesh. Task-oriented tutorial dialogue: Issues and agents. In Rosé and Freedman [2000], pages 52 – 57.

[Rosé and Freedman, 2000] Carolyn Penstein Rosé and Reva Freedman, editors. *Building Dialog Systems for Tutorial Applications—Papers from the AAAI Fall Symposium*, North Falmouth, MA, 2000. AAAI press.

[Rosé *et al.*, 2001] Carolyn P. Rosé, Johanna D. Moore, Kurt VanLehn, and David Allbritton. A comparative evaluation of socratic versus didactic tutoring. In Johanna Moore and Keith Stenning, editors, *Proceedings 23rd Annual Conference of the Cognitive Science Society*, pages 869 – 874, University of Edinburgh, Scotland, UK, 2001.

[Siekmann *et al.*, 2002] Jörg Siekmann, Christoph Benzmüller, Vladimir Brezhnev, Lassaad Cheikhrouhou, Armin Fiedler, Andreas Franke, Helmut Horacek, Michael Kohlhase, Andreas Meier, Erica Melis, Markus Moschner, Immanuel Normann, Martin Pollet, Volker Sorge, Carsten Ullrich, Claus-Peter Wirth, and Jürgen Zimmer. Proof development with ΩMEGA. In Andrei Voronkov, editor, *Automated Deduction — CADE-18*, number 2392 in LNAI, pages 144 – 149. Springer Verlag, 2002.

[Tsovaltzi and Fiedler, 2003a] Dimitra Tsovaltzi and Armin Fiedler. An approach to facilitating reflection in a mathematics tutoring system. In *Proceedings of AIED Workshop on Learner Modelling for Reflection*, pages 278 – 287, Sydney, Australia, 2003.

[Tsovaltzi and Fiedler, 2003b] Dimitra Tsovaltzi and Armin Fiedler. Enhancement and use of a mathematical ontology in a tutorial dialogue system. In *Proceedings of the IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, pages 19 – 28, Acapulco, Mexico, 2003.

[Tsovaltzi and Karagjosova, 2004] Dimitra Tsovaltzi and Elena Karagjosova. A dialogue move taxonomy for tutorial dialogues. In Michael Strube and Candy Sidner, editors, *Proceedings of 5th SIGdial Workshop on Discourse and Dialogue*, Boston, USA, 2004.

[Tsovaltzi *et al.*, 2004] Dimitra Tsovaltzi, Armin Fiedler, and Helmut Horacek. A multi-dimensional taxonomy for automating hinting. In James C. Lester, Rosa Maria Vicari, and Fábio Paraguaçu, editors, *Intelligent Tutoring Systems — 7th International Conference (ITS 2004)*, number 3220 in LNCS, pages 772 – 781. Springer, 2004.

[Tsovaltzi, 2001] Dimitra Tsovaltzi. Formalising hinting in tutorial dialogues. Master's thesis, The University of Edinburgh, Scotland, UK, 2001.

[Weiner, 1992] B. Weiner. *Human Motivation: metaphor, thoeries, and research*. Sage Publications Inc., 1992.

[Wilson and Cole, 1996] Brent Wilson and Peggy Cole. Cognitive teaching models. In D.H. Jonassen, editor, *Handbook of Research for educational communications and technology*, pages 601 – 621. MacMillan, 1996.

[Wolska *et al.*, 2004] Magdalena Wolska, Bao Quoc Vo, Dimitra Tsovaltzi, Ivana Kruijff-Korbayová, Elena Karagjosova, Helmut Horacek, Malte Gabsdil, Armin Fiedler, and Christoph Benzmüller. An annotated corpus of tutorial dialogs on mathematical theorem proving. In *Proceedings of International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1007 – 1010, Lisbon, Portugal, 2004. ELDA.

[Wu, 1996] H. Wu. The mathematician and the mathematics education reform. In Notices of the American Mathematical Society, December 1996.

[Wu, 2001] H. Wu. What is so difficult about the preparation of mathematics teachers. In National Summit on the Mathematical Education of Teachers: Meeting the Demand for High Quality Mathematics Education in America, November 2001.