

ANALYSIS AND INTELLIGENT SUPPORT OF LEARNING COMMUNITIES IN SEMI-STRUCTURED DISCUSSION ENVIRONMENTS

Andreas Harrer
Universität Duisburg-Essen, Germany

Abstract: Conventional discussion environments provide the technical platform for distributed discussion and collaboration, but apart from some statistical data collected, rarely provide information about the collaborative interactions taking place within the environment or even support the discussion by stimulating the learning setting according to the current situation. In this article we present our approach for intelligent support of groups of learners in distributed web-based discussion environments. First we show how the state of a conversation can be monitored using a classification of the respective contributions and describe the design of a user interface specifically tailored for that end. Then we present an implementation of this type of interface and of a proof-of-concept agent supporting the discussion processes actively. That agent analyses the current state of a conversation and makes contributions according to the current situation.

1. INTRODUCTION

A typical characteristic of intelligent tutoring systems (ITS) is the ability to adapt the system's behaviour in the learning process to the learner's needs and the given learning situation. This is possible, because ITS create learner models as a source for making decisions that influence the learning process. The construction of a learner model is achieved by analysing the user's input to the ITS.

With the spreading of computer networks and both technical and software-related possibilities to build distributed systems (such as wireless networking facilities and middleware platforms like CORBA) in the last years research in learning environments focussed on a type of systems called CSCL-systems (Computer Supported Collaborative Learning) that made it possible to learn together in (spatially or temporarily) distributed groups. These systems at first provided the technical facilities for networked learning, i.e. a distributed learning environment. The analysis of the group work itself wasn't topic of research and therefore the systems were not able to support the learning processes of groups as ITS do for single users.

In I-CSCL (intelligently CSCL) systems or, as we call it, Intelligent Distributed Learning Environments (IDLE) the analysis of learning processes which was typical for ITS is extended also on collaborative and cooperative learning processes in distributed learning communities. With mechanisms for such an analysis of interactions, taking place during group learning, the collaborative processes can be supported intelligently by the system.

An IDLE's potential to support learning processes in groups depends heavily on the richness of the information about learning situations and processes. Typically collaborative activities can be distinguished into *domain-related activities* and *coordination-oriented interactions*. Collaborative modelling tools, such as Cool Modes (Pinkwart et al. 2002) tend to rely more on domain-related activities, such as creating/modifying/deleting domain objects. In these systems information about the learning process is inferred by analysis of domain level activities and states resulting by a stream of activities. A classification for analysis methods in these systems can be found in (Gaßner, Jansen, Harrer, Herrmann, Hoppe 2003). On the other side coordination-oriented interactions take place when learners make plans for solving tasks and coordinating their efforts, which are then conducted as activities on domain level. Discussion Environments such as web-based discussion boards provide an infrastructure for this coordination level, but at the moment have scarce facilities for analysing and stimulating the learning processes taking place in the discussion boards. In this paper we present our approach for analysis of interactions and intelligent support in distributed learning communities using such discussion environments.

2. ANALYSIS OF CONVERSATIONS

For the approach of interaction analysis we base our considerations on speech act theory (Searle 79) and conversational theory (Winograd & Flores

86). Speech act theory classifies the type of utterances according to the purpose, called *illocutionary act*, the speaker has in mind. For example if the speaker wants to bring the hearer to do something, the illocutionary act is called *Directive*, because the speaker wants to direct the hearer. Conversational theory investigates in typical patterns and sequences of speech acts in conversations. These patterns seem to be domain-independent in most cases and therefore are usable in IDLEs regardless of different domains of knowledge. Typically those conversations can be represented as conversational networks as the one shown as a finite automaton in Figure 1. That one is a model of a conversation, where one actor requests an explanation and another actor reacts to that request. At the moment the conversation is in state 6, a situation reached by a request for an explanation by student A (state 1 to 2), a commitment of student B for an explanation (state 2 to 3), and then the explanation of B (state 5 to 6).

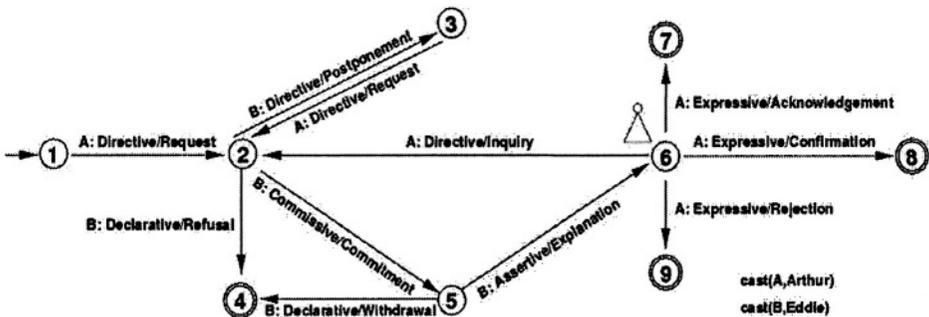


Figure 1: Conversational Network

To use the concept of conversational networks in IDLEs we have to generalize the networks in such a way that they can model interaction between the members of a learning community (usually more than the two participants in conversational networks). Therefore we developed a two-tier-model for interactions on coordination level for learning communities. Our model consists of

- a tier with *conversational networks* each for a pair of participants of the learning community, which represent the individual parts of the interaction; these networks are of the type shown in Figure 1.
- a tier for the community level; represented in a so-called *group network* we describe information at group level explicitly. Events taking place in individual conversational networks can have effect on group level and therefore also on other individual

networks: for example if there is a transition in a individual network as in Figure 1 from state 2 to state 5 the learner who committed herself for an explanation should make her explanation before other students commit themselves also. So at group level we can specify the results of such an event and coordinate the networks for a proper flow of interactions (in this example we could move all the other students into a state, where they cannot commit themselves at the moment, for example state 3). So we have interdependence between individual level and group level.

To use our model of interactions at coordination level for the analysis of interactions taking place in IDLE, we have to find a way to trace the flow of conversations and therefore to diagnose current states at individual and group level. In the next section we present the design of a type of user interface that enables us to get the information needed for this kind of analysis.

3. DESIGN OF A SEMI-STRUCTURED CONVERSATION INTERFACE

Our approach for analysis of conversations is based on the use of a semi-structured interface, similar to the one in (Baker & Lund 96), but not restricted to sentence openers. A complete diagnosis of the content of the messages sent in conversations, is either to obstrusive for the user when using a formal input language or has to use quite sophisticated and reliable methods of natural language processing and text understanding, which are hardly available nowadays. So we chose to have some parts of the message that can be interpreted automatically, which can give us enough information to trace the states in conversations and thus get models of interactions in distributed learning environments.

For our user-interface we have three different kinds of building blocks for a message:

predefined phrases, which have assigned to them specific information usable for learner modeling and that is interpretable for the IDLE

choices, where the user can modify and parameterize the contents; these elements have also some machine-interpretable information in it

free user input, which provides full potential to let the user express herself; these parts are considered unstructured for the further course of this paper and are therefore not machine-interpretable.

Combining these three types of building blocks, we get a more flexible format of messages than in a pure sentence-opener interface, where the user chooses a phrase beginning the sentence and then continuing the message with free input. In Figure 2 we present an example for a sentence template in our semi-structured interface:

**Could you \$Style explain to me \$FillWord \$Keyword
in more detail \$Intensity? \$Problem**

Figure 2: Example for a sentence template in the interface

In this sentence we have all the three types of building-blocks in use. The parts for *\$Style*, *\$Intensity* and *\$Keyword* are choices for the user, where he can choose a more polite form, a degree of urgency (“... explain to me ... in more detail [*now/soon/some time*]...’) and an object (knowledge-unit) the message refers to. The slot with *\$Explanation* is free user input, where he can give details what isn’t clear to him. The rest of the message is a predefined phrase, to which we assign some information about the phrase usable for construction of learner models.

In Figure 3 we give an example for speech act informations assigned to the predefined phrase to give a message a machine-interpretable meaning, if a user chooses that phrase. In that way the user does not have to classify his message explicitly but the classification can be done implicitly without burdening the user with details of the used conversational theory.

Could you ... explain to me ... in more detail?
Category: Directive
Subcategory: Inquiry

Figure 3: Predefined phrase with speech act informations

Now we give a detailed example for the kind of information regarding the interactions in learning communities that can be produced with the help of our semi-structured interface: We assume a learning community with the two learners Arthur and Eddie taking place in a conversation represented in our conversational network in Figure 1. The current state for conversation is state 6, actor A is Arthur and actor B is Eddie. The previous explanation (now called e1) didn’t help actor A to understand his problem, so he chooses the message template in Figure 2 and produces the following message with the help of the semi-structured interface:

“Could you please explain the Markov-Algorithm to me
in more detail?

That is more difficult to understand than I expected.”

The IDLE uses the speech act information of the predefined phrase shown in Figure 3 to trace the state transition in the conversational network from state 6 to 2 caused by the diagnosed speech act (Directive/Inquiry) and constructs an entry for the learner models with information about the interaction that happened by that speech act. This information unit is shown in Table 1.

```

Speech-Act: i1
Sender:      Arthur
Receiver:    Eddie
Category:    Directive
Subcategory: Inquiry
Intensity:   normal
Context:     refers to Explanation e1
Propositional Content:
  (interpreted)  DomainConcept (Markov-Algorithm)
  (uninterpreted) "That is more difficult to
understand than I expected."
Style:       polite

```

Table 1: example for group model content created within the interface

Some of the information is gained directly by the predefined phrase (category, subcategory of the speech act), some by the choices (“please” for a polite style, no specific choice of [now/soon/some time] for a normal intensity) and some by the conversational network (reference to explanation e1). There is also some propositional content in the message that can be interpreted by the system (the choice for *\$Keyword*) and some that is not interpreted, that is the free user input part of the message. We can also make further inferences about that information entry that could give us hints about the roles the two participants have in that conversation. The interactions in that network could imply that Arthur is the student and Eddie is the teacher in the learning community.

The information that we get with our approach of using a semi-structured interface to trace the flow of interactions in learning communities can be a valuable resource for the intelligent support of groups in IDLE (Harrer & Herzog 1999). For example it can be used to diagnose special situations in the group work, where an artificial intelligent tutor gives some hints because the participants are stuck with the problem. The information in learner models can also be used to support the formation of proper learning groups as the approach of Opportunistic Group Formation (Ikeda, Go, Mizoguchi 1997) suggests. For that end the information gained with our interaction analysis can be further processed by other components of IDLE to make intelligent support adapted to the current learning situation feasible. This processing can be based on the analysis of traces/logfiles of interactions at coordination level.

Our mechanism for interaction analysis combined with domain level editors would produce a trace which can be processed to gain further information of higher abstraction level (Barros & Verdejo 1999). For example we can infer a high level of coordination, when we analyze frequent interaction between learners which produced a common problem solution. A large amount of actions and interaction from one specific learner is a strong indication for high participation in collaborative processes. So our interaction analysis is not only usable for tracing interaction processes but also a resource for processing high level contents of learner models such as effort, participation, coordination, dominance and so on, which can be used to adapt the learning process in IDLEs to the given situation. In future work we plan to bring together the two kinds of user inputs, domain level actions and coordinative interactions together in one integrated learning environment.

4. SPREKON - IMPLEMENTATION OF THE SEMI-STRUCTURED INTERFACE

Our approach of a semi-structured interface for conversations is prototypically implemented in the SPREKON interface: It is a web-based conversation system with group management facilities similar to regular threaded discussion boards, but with the underlying principles for classification of contributions according to speech act categories and subcategories as we presented in the preceding section. The implementation of a client-server architecture was done using the web technologies Java Servlets and Java Server Pages to provide a platform-independent environment with minimal installation effort on users' (clients) side. A screenshot of the english version of SPREKON (a testsite is available under

http://blonskij.informatik.uni-duisburg.de:8080/sprekon_english) can be seen in Figure 4.

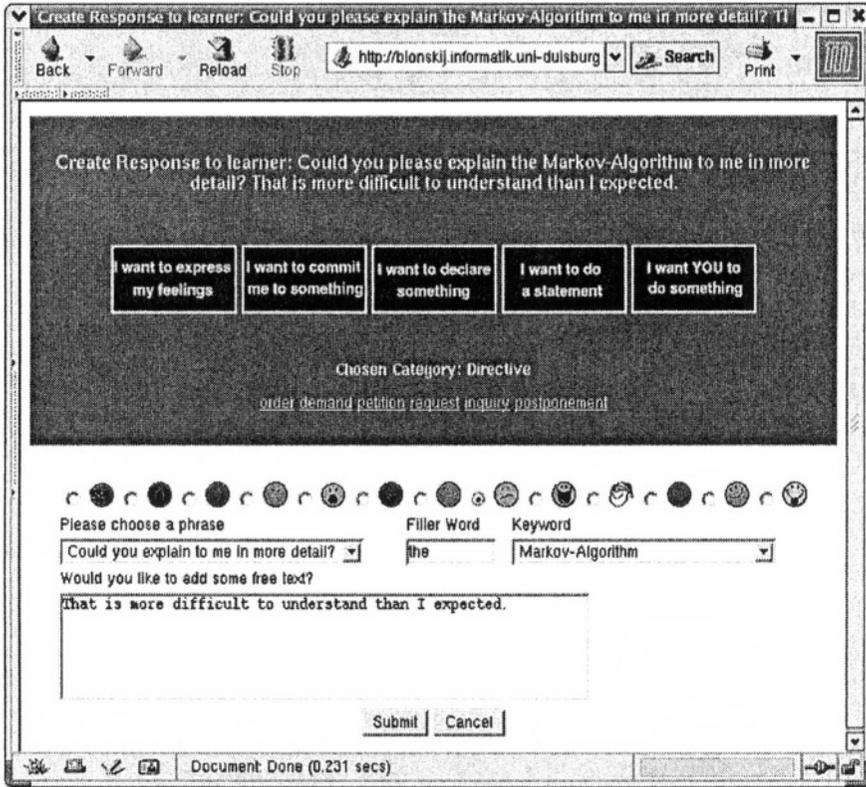


Figure 4: Screenshot of SPREKON interface: creation of semi-structured contribution

In the figure above you can see a window for the creation of a conversational contribution. Following the examples from the preceding section the user has an inquiry to a previously given explanation and therefore chooses a predefined phrase which characterises a directive/inquiry (in the upper part of the window) in speech act categories. Since he specifically has problems with the domain concept Markov-Algorithm, he chooses that keyword from the domain glossary (a choice list on the right side) and finishes his contribution by formulating the problem in a free text area (on the lower part). Other attributes of the resulting speech act can be inferred from the use of respective emoticons and similar UI elements. When the user submits his contribution to the SPREKON-server the contribution is integrated into the conversation thread and internally stored as an XML-document complying to the format demonstrated in the previous section with

some augmentations, such as time stamp, unambiguous references to previous contributions. A simplified example for such an XML-document can be seen in Figure 5.

```
<message id="46711293231602">
  <name>
    <username>Arthur</username>
    <account>arthur@somewhere.org</account>
  </name>
  <date><day>8</day></date>
  <content domain="basicCourse">
<phrasertext>Could you $ explain to me $ in more
  detail?</phrasertext>
  <fillerword>the</fillerword>
  <keyword>Markov-Algorithm</keyword>
  <freetext>That is more difficult than I expected.
  </freetext>
  </content>
  <speechAct category="directive">
    <subcategory>inquiry</subcategory>
    <phrase>Could you explain that in more detail?</phrase>
  </speechAct>
  <reference>
    <predecessor>46759182731502</predecessor>
    <successor>not yet available</successor>
  </reference>
</message>
```

Figure 5: XML document representing a SPREKON contribution

SPREKON is designed for flexibility of application domains, underlying conversational theories, and components interacting with the interface, e.g. for analysis of conversations and intelligent support:

- the glossary of the terms within the specific application domain SPREKON should be used for can be exchanged easily by using another text-based glossary description
- predefined phrases can be exchanged and accustomed to suit the users needs
- categories of the contributions can be exchanged according to the conversational model used (such as Questions-Options-Criteria QOC for handling design rationale conversations)
- the contributions which are stored as XML-files can be used by external applications for analysis purposes, such as statistical

analysis of the types of contributions or more complex analyses (we did this in a classroom experiment and computed collaboration features of a discussion inspired by the procedure in (Barros & Verdejo 1999)).

5. ANAKON - AGENT-BASED SUPPORT OF CONVERSATIONS

To show the potential of intelligent support in learning environments possible with our conversational analysis approach, we designed the ANAKON framework for integration of artificial agents into our discussion environment. These agents can take specific roles for the learning community (Harrer 1999), such as co-learner or learning companion (Chan & Baskin 1988), mediator or observer.

The main components of such an agent are:

- conversational models representing, which high-level actions are appropriate to the current state of the conversation
- detailed plans how the high-level actions can be conducted in atomic steps
- domain knowledge base, if the agent's task is to contribute to the domain level discussion (this would not be necessary for a mediator agent exclusively responsible for the flow of interaction)
- building blocks to create the text of the agent's contributions

As proof-of-concept for our agent-based approach, we implemented a learning companion agent that has some kind of domain knowledge and thus in some situations can provide helpful domain level comments and in some cases cannot, a typical form of a learning companion with "mediocre abilities". We parameterized this agent in respect to the components from above as following:

- the agent has as conversational model the typical conversational network for requests, explanations, and inquiries as shown in Figure 1, where the agent is taking the role of actor B
- detailed plans for each action the agent may take in the conversational model, for example before committing to an explanation the agent first checks, if he has information about a keyword useful for the explanation, and if it has information it selects a text phrase for the commitment. A plan for this short example is specified in an XML document like this:

```

<commitmentPlan>
  <action>lookupKeyword</action>
  <phrases>
    <text>That is easy, listen:</text>
    <text>I will explain that to you.</text>
    <text>OK, I'll tell you.</text>
  </phrases>
</commitmentPlan>

```

- the agent has a part of the domain glossary with keywords, definitions and explanations to use in its explanations
- the agent has some phrases as building blocks for the contributions but could as well use the building blocks learners use in the SPREKON interface to create the text messages.

An example for a dialogue between a learner (learner) and the agent learning companion (compie) looks like in Figure 6:

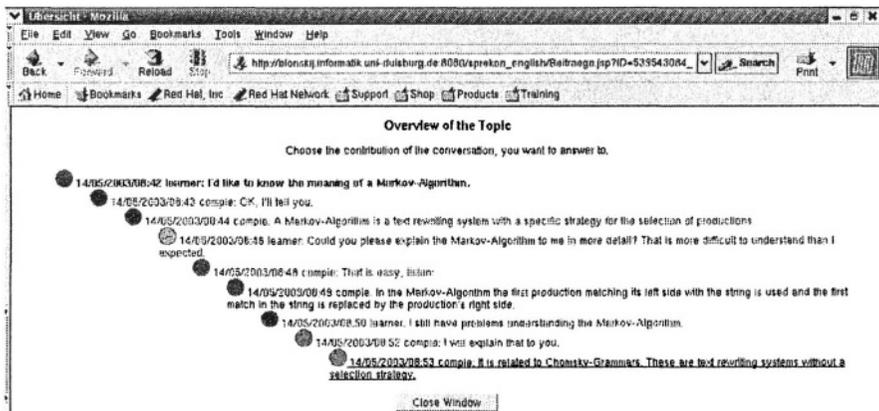


Figure 6: Example dialogue between learner and artificial learning companion

In the dialogue you can find the typical pattern of speech acts (request/inquiry, commitment, explanation) represented in Figure 1. The learning companion twice has some information about the keyword *Markov-Algorithm*, but the third time it just has a crossreference that may or may not be helpful to the other learner. Adding more conversational models and

generic parameterizable phrases in the future will improve the credibility of the agents behaviour.

6. CONCLUSIONS

In this article we presented our approach for analysis and intelligent support of distributed learning communities using discussion environments. The analysis of the flow of interaction in conversations is based on speech act theory and conversational theory. We designed a semi-structured user interface that produces the output needed for this mechanism of analysis and implemented this interface type as the web-based SPREKON conversational interface. This interface has been used in a seminar and in the exercises to a lecture both in computer science at our university. We also described our framework for integration of artificial agents into the discussion environment. These agents can support the collaboration by giving hints, explanations or providing other types of help based on their respective roles. A learning companion agent has already been implemented as proof-of-concept of this support. We plan to test the agent's impact on learning scenarios in future experiments. Recent development within the SPREKON context has been done in a component conducting Social Network Analysis (SNA) (Wassermann & Faust 1994) in the structured discussion threads. With that centrality and specific roles of users in the conversations can be analyzed. In future work we plan to prepare the SPREKON interface for different domains of knowledge to be discussed and to use this interface in a larger scale, e.g. for big classes at university. Another goal we have is to integrate SPREKON with domain-level oriented collaborative tools and to use the rich potential analysis of both interactions and domain activities will give for intelligent support of learners. We also plan to implement other agents (Harrer 2000) in the ANAKON framework which shall take different functions/roles for the learning community.

ACKNOWLEDGEMENTS

My thanks go to my students Margret Bauer, Jörg Geiger, Angelika Müller for their implementation work in SPREKON and ANAKON, and my colleague Clemens Harlfinger for his support in the supervision of these projects.

REFERENCES

- Baker, M., & Lund, K. (1996). Flexibly structuring the interaction in a CSCL environment. In P. Brna, A. Paiva, and J. Self, (eds.). *Proceedings of the European Conference on Artificial Intelligence in Education EuroAIED'96*, Lisbon.
- Barros, B. & Verdejo, M.F. (1999). An approach to analyse collaboration when shared structured workspaces are used for carrying out group learning processes. In Lajoie, S.P. & Vivet, M., (eds), *Proc. of the World Conference on Artificial Intelligence in Education AI-ED 99*, 449-456, IOS Press Amsterdam.
- Chan, T.W. & Baskin, A.B. (1988). "studying with the prince". In Frasson, C. et al. (editors), *Intelligent Tutoring Systems*, Montreal.
- Gaßner, K., Jansen, M., Harrer, A., Herrmann, K., Hoppe, H.-U. (2003). *Analysis Methods for Collaborative Models and Activities*. In *Proceedings of Computer Support for Collaborative Learning CSCL2003*, Kluwer Academic Publishers, Dordrecht.
- Harrer, A. (1999). Group learning in hybrid communities in intelligent tutoring systems. In Lajoie, S.P. & Vivet, M., (eds), *Proc. of the World Conference on Artificial Intelligence in Education AI-ED 99*, 708-710, IOS Press, Amsterdam..
- Harrer, A. (2000). *Unterstützung von Lerngemeinschaften in verteilten intelligenten Lehrsystemen*. PhD thesis, Technische Universität München, Institut für Informatik, 2000.
- Harrer, A., & Herzog, C. (1999). SYPROS going IDLE - from a classical ITS to an intelligent distributed learning environment. In Cumming, G. et al. (eds). *Proceedings of ICCE '99, 7th International Conference on Computers in Education*, 836-839, IOS Press, Amsterdam.
- Ikeda M., Go, S., Mizoguchi, R. (1997). Opportunistic group formation. In du Boulay, B., & Mizoguchi, R. (eds.) *Proceedings of the World Conference on Artificial Intelligence in Education AI-ED 97*, 167-174, Kobe, IOS Press, Amsterdam.
- Pinkwart, N., Hoppe, H.U., Bollen, L. & Fuhrott, E. (2002). Group-oriented modelling tools with heterogeneous semantics. In *Proceedings of Intelligent Tutoring Systems ITS-2002* . Springer, Berlin,
- Searle, J.R. (1979). *Expression and Meaning*. Cambridge Univ. Press, Cambridge, UK.
- Wassermann, S., & Faust, K. (1994). *Social Network Analysis: Methods and Application*. Cambridge University Press, Cambridge.
- Winograd, T., & Flores, F (1986). *Understanding Computers and Cognition*. Ablex Publishing Corporation, Norwood, New Jersey.