

Designing a Multi-agent Architecture based on Clusterization for Collaborative Learning Sessions

Emmanuel Blanchard, Claude Frasson

Université de Montréal, Département IRO

C.P. 6128, Succursale Centre-Ville, Montréal (Québec), H3C 3J7 Canada

E-mail: {blanchae,frasson}@iro.umontreal.ca

Abstract : collaboration between learners can have a positive impact in a learning session only if learners can exchange efficiently. Discussions, advices given by a co-learner are good means to help a learner in knowledge understanding. One of the main problems in a collaborative learning activity is to create coherent groups among students. In this paper, we present an architecture that aims to address this question. We use a neural network algorithm to obtain homogenous groups. An architecture based on several agents is then able to select the more relevant group and provide the learning session with information related to the subject.

Keywords : collaborative learning, student categorization,

1. Introduction

In practice, what how a learning session is organized? Do students work alone or is there an advantage having workgroups? Many research about collaborative learning tend to prove that best results are obtained giving students some tasks to perform in group. Adams and Slater (2001) emphasize the fact that students are more involved in their own learning and learn more from a discussion with a fellow than by following lectures. Gokhale (1995) argues that “collaborative learning enhances critical thinking”. Kapitzke (2000) notices that, during a learning session, a large part of students will spontaneously work in pairs or groups (more than 90%) and feedbacks (referred to collaborative work) from those ones were mostly very positive. Discussion and communication appear to be major aspects of learning.

Those results can be extended to collaboration with “Computer-Mediated Communications” (CMC). Mc Ateer et al (1997) portrayed CMC, in the distance education domain, “as an ideal means for providing opportunities for group discussion, student-centered interaction and collaborative tasks”. But, as Adams and Slater (2001) say, one of the main problems in collaborative learning is to find a good way for creation of learner groups.

In this paper, we present a multi-agent architecture for supporting collaborative learning. We use a strategy based on non supervised neural network, developed in Alice/Whiterabbit system (Thibodeau et al, 2000), to build homogeneous groups of learners. We also propose an option to enrich knowledge base while learning sessions are occurring, using the huge resource contained in the World Wide Web.

2. Overview of the Alice and White-rabbit system

This system aims at upgrading communications inside a company, matching users interested in a specific knowledge with those that master effectively the knowledge. Many other matchmaking agent-based systems exist: Yenta (Foner, 1997) or Butterfly (Van Dyke et al, 1999) for example.

In the Alice/Whiterabbit system, user profiles are elaborated using keywords recognition (figure 1). The user profile contains two sets of information : *interests* of the user detected from his communications with other learners (email, chatroom), and *knowledge level* which is deduced from the analysis of documents written or produced by the learner (reports, communications, memos). A degree of certainty is associated to each deduction.

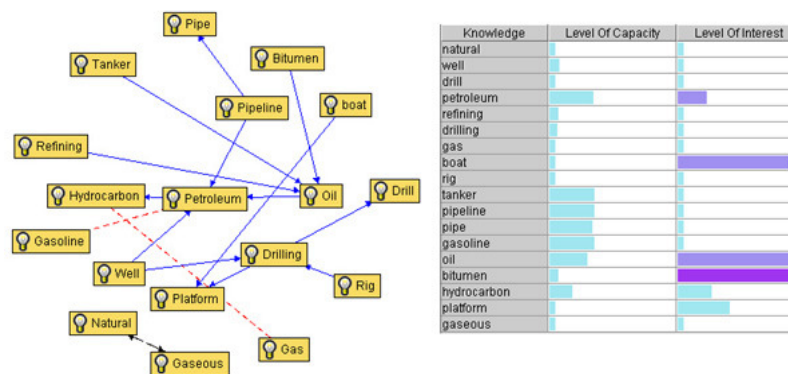


Figure 1 : a keyword graph and profile representations in the Alice/White-rabbit system

Keywords are organized in a graph including dependencies between terms. When a keyword is detected the links are taken into account to update profiles. Once the system has enough information, it can cluster people, grouping users according to their profile. This operation is realized using a unsupervised neural network called “SOM” for Self Organizing Map (Kohonen, 1990). With such groups, the system is then able to make connection between complementary people. With Alice/Whiterabbit, we obtain clusters

of people having same interests or the same knowledge level. This important point joins the objective of an efficient collaborative learning we mentioned above : creating homogeneous groups for a Collaborative Learning Session (CLS).

3. An Agents-based Architecture for Supporting Collaborative Learning

The cluster method of Kohonen SOM, combined with well defined user profiles, appears to provide interesting groups of learner. We propose and explain an architecture (figure 2) which integrates the first method to improve a CLS.

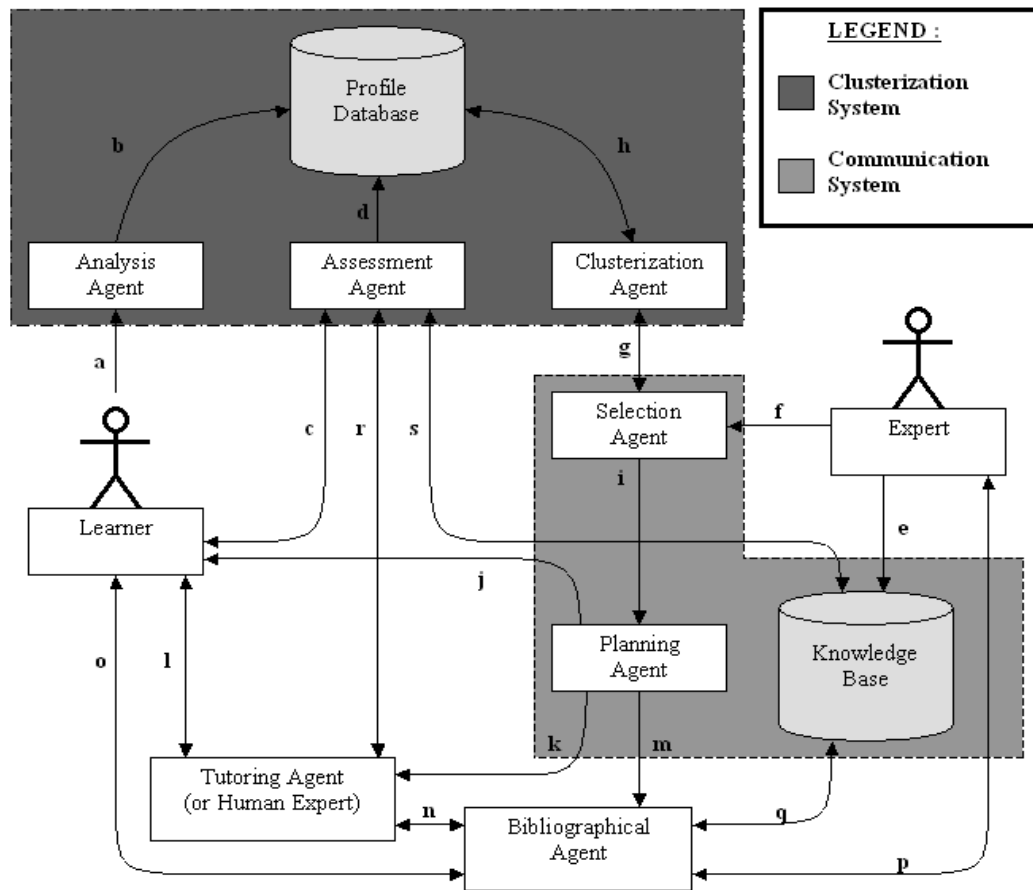


Figure 2 : Architecture of the System

The architecture is divided into two main components: (1)the first one is derived from the Alice/White-rabbit architecture and produces a clusterization schema both of

learners interests and knowledge, (2) The second one use the clusterization schema to establish adequate communications between learners.

3.1 The clusterization component

An *Analysis Agent* “analyzes” discussions (a) of a learner with others to find his interests through analysis of emails and chat exchanges using a list of predefined keywords and updating the learner interest profile (b) in the profile database which contains the profile of all learners.

The *Assessment Agent* completes learners profile database (d) with information related to his knowledge level by processing documents produced by the learner (c).

The *Clusterization Agent* extracts information from learner profiles (h) in order to present different clusters of learners distributed according to their interest or their knowledge level.

This component is fully implemented and has been tested. The system is presently distributed by Virtuel Age International, a company specialized in knowledge communication and learning.

3.2 The Communication Component

This part is created and maintained by a subject matter expert in order to provide the best (which means the most useful) communication between learners. This one intends to initialize a learning session between different learners. In order to find these learners, he communicates a list of criteria to the *Selection Agent* (f), such as : the knowledge level required for the session and the degree of interest and knowledge variance. The selection agent communicates with the *Clusterization Agent* (g) in order to dynamically extract an instance of the clusterization schema adapted to the session. This

set of learners is then passed on to the *Planning Agent* (i) that will initialize the virtual classroom by inviting selected users to participate to the session (j). The agent will also send to the *Tutoring Agent* a list of information including : the list of learners with their respective knowledge level and a predefined decomposition of the subject matter corresponding to these conditions (k). The *Tutoring Agent* that can be either a system or a human will be able to provide teaching with these adapted information for a collaborative session (l). He will then ask the *Assessment Agent* of the clusterization component to evaluate the new level of knowledge of learners by testing their capabilities (r). Such tests can include multiple choice queries, team games ...

Another agent is involved in our architecture : the *Bibliographical Agent*. Its role is to find relevant information on the Internet. It can receive search request from the *Planning Agent* (m), the *Tutoring agent* (n) or from a learner (o). Once interesting resources are found, the *Bibliographical Agent* will submit its results to the expert (p). If the content is judged to be pertinent, they will be added to the knowledge base (q). This verification is useful in order to provide information adapted to the right class of learners.

4. Operational Aspects

After this overall view of the architecture, we focus on some specific aspects and roles of our agents.

The *Selection Agent* provides an interface to the expert in order to help him to find the best group of learners for the course he wants to provide. Figure 3 is an example of such an the interface. Each predefined keyword is a possible parameter. The expert can decide about the minimum knowledge or interests the learners should share. He can

check and establish parameters he wants to use and set it to the corresponding value. He can also limit the variance level of those parameters inside a cluster in order to obtain an homogeneous group of learner. The agent will then transmit directives of the expert to the *Clusterization agent* who will retrieve the best fitting cluster.

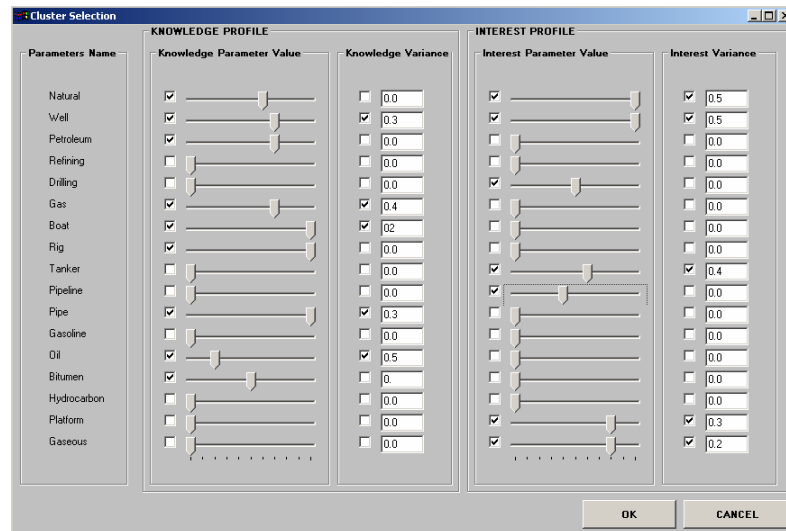


Figure 3 : an Example of Interface for the Selection Agent

Figure 4 is an example of such query. Each parameter is described by the profile it belongs to, its name and the type of information (value or variance) that will be tested with the limit value. The best cluster will include the higher number of successful conditions. Finally, when the cluster is chosen, and transmitted to the *Planning Agent* that which will deal with the *Tutoring Agent* in order to provide a complementary learning session to improve the level of learners.

```

Find CLUSTER where
{
    KNOWLEDGE.bitumen.value > 0.9
    KNOWLEDGE.platform.value > 0.8
    KNOWLEDGE.gas.value > 0.9
    KNOWLEDGE.pipeline.value > 0.8
    KNOWLEDGE.pipeline.variance < 0.3
    KNOWLEDGE.pipe.value > 0.8
    INTEREST.boat.value > 0.5
    INTEREST.boat.variance < 0.2
    INTEREST.gas.value > 0.8
    INTEREST.gas.variance < 0.2
    INTEREST.drilling.value < 0.3}

```

Figure 4 : an Example of Query from the Selection Agent

The *Bibliographic Agent* is also very important because he is in charge of the evolution of the knowledge base. In our model, it hasn't been yet implemented but we suggest to use techniques like as described by Somio and Howe (2001). They gather information using a method they called doubling incremental clustering which provide good results in text filtering (that algorithm could also have a great impact in the Alice/Whiterabbit System).

The *Tutoring Agent* will be facing new challenges. How to manage a CLS and take the best advantage of the collaborative aspect? The way courses will be dispensed will have a great impact on collaboration between co-learners. The tutor will have to promote communication, exchanges... Chat room is the basic step. Studies show that virtual reality could be another good point (Chee, 2001). We can also use multimedia pedagogical resources made with the help of intelligent agents like the Intelligent Classroom (Franklin, Hammond, 2001).

CLS will also make new exercises possible. Although classical assessment methods will be still available, we could for example develop a Trivial Pursuit-like game where learners will have to compete, once again to enhance collaboration and exchange. Game and competition are always good to keep the attention of an audience.

5. Conclusion

We have presented a multi-agent architecture, based on the principle of clusterization of a population of learners, in order to create homogeneous groups to provide CLS. We also joined a web-searching agent, making our knowledge base evolving dynamically and becoming more rich lessons after lessons. Collaborative

Learning is a very interesting and few exploited field in Computer Assisted Learning. We think our architecture can give good results given the fact connected people have similar profile. This should enhance communication and emulation inside a group. Actually, many aspect of a human-to-human discussion aren't well synthesized.

In a society where more and more people access to high level studies, where information is doing a new revolution and where people can communicate all around the world, Collaborative Distance Learning may be a key for new learning methods.

6. Future Work

Presently, only the clusterization system has been implemented. Email support for the Analysis Agent has been implemented in order to run in a company with the most commonly used email system. The Selection and Planning Agents are currently implementing while the interface with the expert is completed. We are examining the best way to store pedagogical contents for enhancing collaboration between learners. We will proceed by evaluating the contribution of each part of this architecture. The first one has already been tested as useful for showing a clear view of knowledge distribution in a class or in a company.

7. References

Adams J.P. and Slater T.F. (2001), "*Implementing In-Class Collaborative Learning Group Activities in Large Lecture Astronomy*". Journal of College Science Teaching, fall 2001.

Chee Y. S. (2001), "*Networked Virtual Environments for Collaborative Learning*", Invited talk, In Proceedings of ICCE/Schoolnet 2001- Ninth International Conference on Computer in Educations, Seoul, 2001.

Foner L. (1997), "*YENTA: A Multi-Agent, Referral-Based Matchmaking System*". In Proceedings of the First International Conference on Autonomous Agents, Marina del Rey, 1997.

Franklin D. and Hammond K. (2001), "*The Intelligent Classroom: Providing Competent Assistance*". In Proceedings of the Fifth International Conference on Autonomous Agents Montreal, May 2001, pp.161-168.

Gokhale, A. A. (1995), "*Collaborative learning enhances critical thinking*". Journal of Technology Education, 7(1) , 1995

Kapitzke, C. (2000). "*The Sociality and Spatiality of Online Pedagogy and Collaborative Learning in an educational media and technologies course*". Educational Technology and Society, July 2000.

Kohonen, T. (1990), "*The Self-Organizing Map*". In Proceedings of IEEE, vol. 78, n° 9, 1990, pp.1464-1480.

McAteer, E., Tolmie, A., Duffy, C., and Corbett, J. (1997), "*Computer-mediated communication as a learning resource*". Journal of Computer Assisted Learning, 13(4), 1997, pp. 219-227.

Somio G. L. and Howe A. E. (2001), "*Incremental Clustering for Profile Maintenance in Information Gathering Web Agents*", In Proceedings of the Fifth International Conference on Autonomous Agents Montreal, May 2001, pp.262-269.

Thibodeau, M. A., Bélanger, S, Frasson, C. (2000), "*WHITE RABBIT - Matchmaking of User Profiles Based on Discussion Analysis*". In Proceedings of the International Conference in Intelligent Tutoring Systems, Montreal, June 2000, Lectures Notes in Computer Science, Springer Verlag no 1839.

Van Dyke N. W., Lieberman H. and Maes P. (1999), "*Butterfly: A Conversation-Finding Agent for Internet Relay Chat*". In Proceedings of International Conference on Intelligent User Interfaces, Redondo Beach, January 1999.