Learning IS Active :An Intelligent Social Learning System

Jiah-Shing Chen, Gwo-Dong Chen, Jong-An Lai, Jen-Kai Liang,
Chi-Zen Lin, Shi-Jen Lin, Baw-Jhiune Liu, Tak-Wai Chan

Institute of Computer Science and Electronic Engineering
National Central University
Chung-Li, TAIWAN 32054
R. O. C.

Abstract

This paper describes the design framework and some development issues of the project, LISA (Learning IS Active), which uses a network of computers to teach freshmen for the course "Introduction to Computer Science". The research is inter-disciplinary in nature, comprising computing, multimedia, artificial intelligence, cognitive science, computer network, etc. For example, different medium information provides multiple learning cues for the students while techniques of artificial intelligence allow the system to better adapt to individual student's needs. A novel element of this research is to engender student's initiatives or motives to learn by engaging the student in a learning environment with rich social context. Some pilot studies of the project have been conducted.

1. Introduction

There is an old Chinese adage stating that the *Five Educational Objectives* are *Morals*, *Knowledge*, *Athletics*, *Sociability*, and *Art*. Among them, knowledge is getting so important that one can financially survive in a modern society depends on what and how much one can learn. To meet the demand of the society, formal schooling which only emerged in the last century has become a prevalent method of education. But we know that education is highly labor intensive. School teachers are heavily burdened by their role of transferring knowledge to students — so heavily that they rarely find enough time to pay attention to other educational objectives, not to mention the students' individual needs. In fact, when schools function as factories in order to produce massive graduates equipped with the needed knowledge for institutions such as commerce and industry , student's knowledge acquisition process in schools remains ineffective, nor has it fully benefited by modern technology.

LISA (Learning IS Active), a project undertaking at National Central University, Taiwan, intends to build an intelligent multimedia learning environment with particular stress on social cognition. LISA has two objectives:

(i) To simulate a futuristic high-tech learning environment

LISA is a distributed system connected with workstations. Each workstation is located at a small room, used by one or two students and the system is used to teach freshmen the course "Introduction to Computer Science" . Furthermore, technology of artificial intelligence is adopted to detect and react student's learning performance and motivational state so that the system can be more adaptive to individual student's need. Thus, we may simulate a futuristic intelligent learning environment where a student at home can learn together with one or more of his/her neighbors and communicate via network with another small group of students at a different geographical location.

(ii) To use multi-channels to promote students to learn actively and effectively

By multi-channels, we mean apart from receiving multiple learning cues from different media[1], students are involved in interpersonal interactions. Since a computer can be simulated as a teacher and/or peers of the students, and each student group can cooperate with or compete against another group, students are bound to interact actively with various agents in the process of learning. With the higher density of learning information generated and exchanged in such an environment, students' motivation to learn is aroused.

This paper is organized as follows. In the next section, we give a background of the project. Some pilot studies which are conducted before we begin the current project are briefly discussed in section 3. In section 4, a learning model called OCTR which captures student's social cognition at different learning stages and provides a conceptual framework for designing the system is described, followed by a quantitative explanation of the model in section 5. We report the current status of LISA

---

[1] By the turn of twenty-first century, we may envision that multimedium computer aided instruction (CAI) will be a widespread means of acquiring new knowledge and training, just as the current use of video tapes for entertainment.

0-8186-3730-7/93 $3.00

in section 6, and give the concluding remarks in the last section.

2. Background

Private tutoring has proven to be as much as four times as educationally effective as a normal classroom setting and 98 percent of the students perform better with private tutors [Bloo 86]. The poor students are benefited the most. Intelligent tutoring systems (ITS), modeled after the idea of a private tutor, provide individualized instruction which offers the potential of better attention to individual student needs. This original vision remains strong today [Clan 86].

Realized as an interactive computer program, most ITS naturally can be conceived as a two-agent model — a learning environment with a *human student* and a *computer simulated teacher*. However, Gilmore and Self [GiSe 88] argued that the role of computer as an authorized teacher for transmitting certified knowledge should be de-emphasized. Furthermore, Self and his colleagues [GuSe 90, DiSe 91, DiSe 92, GiSe 88, Self 85, Self 86] suggested that the computer can be treated as a collaborator. Chan and Baskin [ChBa 88, ChBa 90], as another avenue, proposed that computer can be simulated as two coexisting agents, a teacher and a companion, rather than the traditional single teacher oriented ITS. The two agents, the computer teacher and the computer companion, together with the human student, form a three agent learning environment which is called the *Learning Companion System* (LCS) . The companion in an LCS can act as either a collaborator or a competitor at different learning stages. Chan [Chan 91] has reported an LCS prototype in the domain of integration calculus. The general trend of this research is trying to recover the social context that have been lost in most computer-based learning environment.

Psychological foundation of this research has been laid by various studies on individual cognitive development under social interactions. Piaget [Piag 67] regarded peer interactions as an ideal forum for helping children shift away from egocentric views, enable them to consider multiple perspectives, foster a more comprehensive mature conception to emerge, and thus take a leap to a higher level of understanding. Vygosky [Vygo 78], the developmental theorist who most emphasized the social nature of cognition, argued that cognitive development process is the gradual internalization and personalization of what was originally a social activity.

On the other hand, rather than the cognitive process, some research has focused on motivational aspect of social learning. For example, cooperative, competitive, and individual incentive structures are compared and contrasted [Slav 83, JoJo 74, 75]. Motivation, a factor to determine whether the student *will* do, not just whether he/she *can* do, is certainly desirable in a learning environment to enhance a student's achievement-striving behavior. Indeed, the society is basically forming from collaborators and competitors. But whether it is in collaboration or competition, the mere existence of a peer may constantly stimulate discrepancy which arouses the student's need of competence.

3. Some Previous Studies

We have conducted three pilot studies prior to undertaking the project. The first one is to implement and evaluate a distributed learning companion system (DLCS) on a simple domain which consists of two connected computers so that students can learn in collaboration and/or competition at different locations [Ch+ 92]. The second one is to design a knowledge-based architecture for the implementation of intelligent tutoring systems [Chan 92]. The final one is to design and test a coherent set of highly simplified programming languages (including Lisp and object-oriented programming languages) as the teaching material [ChWa 92]. These languages have been used in two freshmen classes at National Central University and the seemingly difficult teaching goal is proved to be achievable for the freshmen.

3.1 Distributed Learning Companion System

As indicated in [ChBa 90], students can learn together through geographically distributed networks of computers, perhaps without a human teacher, to form an *intelligent futuristic computer classroom.* A computer aided instruction system classic, WEST, is re-implemented but the version is a distributed one in which two students can play concurrently at different locations to learn manipulation of binary numbers. We found that there are many possible models of learning in distributed WEST system based on different combinations of agents and dimensions of factors such as roles, number, and levels of the agents. For example, a computer teacher, if exists, can be a tutor, a coach, a critic, or an evaluator. A computer companion, if exists, can be a collaborator or a competitor of the human student. Some students prefer learning with more advanced students for being more challenging and able to learn from better performance, and some do not prefer doing so for avoiding pressure and being nervous. Likewise, some students would like to place themselves out of sight of their learning companions or do not want their companions to know who they are, because they may feel humiliated for their faults or slow response. Considering all of the factors that may affect the learning result, we have enumerated 768 prototypes which is the size of the Cartesian product of those dimensions (not orthogonal) of factors.

Given the vast number (768) of possible models of the DLCS system, how to choose significant prototypes to be evaluated is a problem [ChCH 92]. Nevertheless, the learning effects of DLCS in a preliminary field evaluation is positive and encouraging. Most students think that it is effective to learn with a companion. To our surprise, we found that students seem to prefer competition to collaboration. This leads to two implications. First, for a long time, a tremendous amount of ITS research has been directed to the student modeling for better adaptability to a student, but only based on the student's cognitive behavior, while initiatives or motives of the student has not been seriously studied [Sold 92a, 92b]. Second, while competition provides a powerful motivation and organizer of learning as well as a strong focus for students' attention and improvement, among computer aided instruction (CAI) and ITS communities, there is a phobia of competition. This phobia seems to be the extrapolation of the social blame that competition has been a means of distributing educational resources to the young; for example, using public examinations to screen students. Contrary to this pessimistic attitude, ITS research should not ignore the invaluable learning motives and striving attitude engendered by competition and try to avoid the ill effects, caused by revealing strengths and weaknesses in comparison, probably by nurturing collaboration simultaneously. Indeed, learning quality under a competitive environment, perhaps, in accordance with Darwinian evolution theory, elevates.

3.2 A Knowledge-Based Architecture of Intelligent Tutoring Systems

Social learning activities can be described by three levels of abstraction. The global level is the *curriculum level*. The curriculum is the whole discourse of learning activities in a certain structure. The second level is the *protocol level*. A protocol organizes learning activities in a certain format. The last level is the *episode level*. An episode is a basic unit of learning activities which usually has a beginning and an end. An episode can also be viewed as an instance of a protocol.

In short, a curriculum consists of a set of protocols, a protocol is composed by a set of episodes. In other words, protocols and curriculum are abstract descriptions of episodes. We can represent these three levels of abstraction by a tree structure, called *curriculum tree* [Chan 92]. In terms of this curriculum tree structure, the curriculum is the whole tree or the root node of the tree, the protocols are the internal nodes above the episode nodes which are the leaf nodes.

The curriculum tree is essentially a tree of knowledge-based systems working together based on the domain structure. It can be viewed as a two-level blackboard architecture. At the local level, the episode node is a blackboard system where several agents in a learning society, represented by production systems, are the knowledge sources. All behavior will then be determined by the rules of behavior including those modeling a protocol of activities. The rules of behavior may in turn call some supporting functions. At the global level, every single episode node is a knowledge source. During scheduling for selecting episode, a scheduling node in an internal node of the curriculum tree is activated. This scheduling node, a production system by itself, is the scheduler. The scheduling strategy is defined by the scheduling rules of the node and the data on the blackboard. This view provides the conceptual separation between managing a tree of a knowledge-based system and running a particular knowledge-based system.

Another view can be based on the language constructs that have been built on top of the LISP language. The curriculum tree is a higher level language for describing a particular curriculum structure of learning activities. These learning activities can be organized in terms of the learning goal hierarchy which can be seen as a mapping on the domain knowledge conceptual dependency structure. The basic entity of the code for learning activities is a rule of behavior of an agent, and for scheduling the basic entity is a scheduling rule. These basic units of code are produced from another higher level language, the production system. All these rules are organized in the curriculum tree. The system runs locally by the inference engine of the blackboard system in an episode and globally by the inference engine of the scheduling node in the curriculum tree.

Primarily based on the subject domain knowledge structure, curriculum tree naturally incorporates the global curriculum planning and monitors the local learning activities. By adopting rule inheritance, the architecture allows additional additivity and flexibility for developing an intelligent tutoring system incrementally as well as efficiency for running rules in each learning episode. We are currently exploring the entension of this architecture in order to equip the system with student modeling ability and capture the node-link model of the hypertext style program for the leaf nodes of the curriculum tree.

3.3 Design of Multiple Pedagogical Programming Languages for Freshmen

To keep up with the rapid changes, fundamental computer science education should introduce the modern concepts of the discipline to the young as early as possible in their computer education. The theme of the first course of computer science should try to make the students appreciate the power of computer in expressing and solving problems of various types flexibly. *Learning multiple programming languages* is a good means to introduce contemporary computer science concepts.

A reasonable student is a good problem solver when equipped with an appropriate problem solving model for the given problem at hand while a programming language usually promotes a certain problem-solving model. That means when the student understands the current problem, he/she can easily map the problem to the problem-solving model supported by the given programming language, obtaining a mental schema of the problem hazily represented in that language before supplementing with detail to obtain a program.

We have designed a set of small languages (interpreters) written in Scheme [AbSu 85] and have been used as teaching material for two classes of freshmen, Fall, 1991 and Spring, 1992. Preliminary evaluations after the courses show that the effect of learning these languages is satisfactory. Both classes learnt six languages in the order Core, Block, Lisp, ADT, Prototype, and SOOP. Core is comprised a few basic elements of most high level programming languages for the consideration of beginning to learn programming. Block can be viewed as a subset of Pascal by teaching block structure via nested function declarations as well as procedural abstraction, structured programming, and top-down design. Lisp is a tiny subset of Common Lisp. We teach recursion and the 'large' data type, list. The last three languages are designed for teaching the prevalent object-oriented programming. ADT which imitates CLU [LiAS 77] is designed for teaching data abstraction. Prototype is a simplified language described in [Lieb 86] for presenting the concepts of objects, delegation, and message-passing. SOOP is a Small Object-Oriented Programming language which can be viewed as a combination of ADT and Prototype.

4. A Social Learning Model

OCTR is a social learning model which is inspired from the well-known apprenticeship learning model. Admitting the ineffectiveness of formal schooling, some researchers resorted to the apprenticeship learning which was the most common means of learning before schools appeared [CoBN 89]. Apprenticeship learning model resembles the skill acquisition process of an apprentice from a master, and is basically composed of four stages [PaSe 88]:

Modeling: where the student observes how the teacher carries out a task repeatedly;

Coaching: where the teacher helps or gives hints to the student to accomplish a task;

Fading: where the teacher's help is reducing gradually when the student is being able to perform the task; and

Reflection: where the student observes his/her own learning process and compare with others.

Modeling is to help the student to build a conceptual model for the whole task. Coaching is an alternating sequence of scaffolding and fading process. Here scaffolding refers to the support by a teacher who carries out parts of the overall task that the student cannot yet managed. An example is that, at the beginning of learning writing, a parent holds a child's hand to help. Another form of scaffolding is offering advice, hints, critics, etc., by the teacher during the process. Scaffolding may also be a learning tool, for instance, a baby's walker. In a computer-based environment, the aiding tool can be a microworld where the student performs parts of the task and the rest is consummated by the system. Fading begins when the need of the teacher's support is not indispensable. But when the student is moved on to a more complex task, the same amount of scaffolding will then be resumed. Reflection enables the student to compare with an expert or another student and usually involves various techniques or tools for reproducing the past performance of the student. An example is the discussion between a tennis coach and a trainee during the video replay of the trainee's playing process.

While apprenticeship learning model accurately describes how a master applies different teaching strategies at different stages in the learning process of an apprentice, it can be extended in a richer social context, mimicking the intensive knowledge communication nature of the modern society. OCTR model, like apprenticeship learning, views learning as incrementally staged process. In these stages, learning objectives are defined and teaching strategies are incorporated. But unlike apprenticeship learning model, social context is incrementally embedded to motivate the students to learn. OCTR, again, composed of four stages:

Orientation: in this first stage, the system helps students to relate their prior knowledge of the domain;

Coaching: in this second stage, the teacher helps students to handle the task on their own mainly by scaffolding and fading, like the apprenticeship learning at that stage;

Tuning: in this third stage, students should have acquired a basic knowledge or skill to tackle the task independently. Knowledge restructuring is exercised via peer collaboration and discussion and the teacher takes a less active role; and

Routinization: in this last stage, students solidify their knowledge by keeping practice under peer pressure.

Relating the student's prior knowledge of the subject is the primary objective of the orientation stage. As the prelude of a piece of music, it helps students to go into the subject with better attention. There are various ways to achieve orientation. We may give a preview and then goes into the detail directly by a description of the material to students, as many teachers do in class. Unfortunately, this approach tends to maintain the subject at an abstract level that a beginner finds it difficult to conceive a new concept. Another way is to choose carefully examples in real life and organize the material in a story form in order to recall the students' experience, arouse curiosity, and enable them to grasp the concept initially with an analogical model. Perhaps the most preferred way at that stage is discovery learning where students explore and gain experience on their own in a simulation game or microworld.

Coaching is to help the student to establish an initial mastery skill for a task. To be able to accomplish a task, at the beginning, modeling or demonstration is needed to externalize the cognitive processes or activities for capturing a conceptual model. After that, the basic scheme of interaction is to apply scaffolding and fading alternatively. Scaffolding is to ensure learning within the student's capacity while fading forces the student to infer more detail from what they have already known. Meantime, reflection is employed by replaying and reviewing the student's learning process and achievement. As can be seen, a large part of apprenticeship learning has been squeezed into this stage.

As tasks are sequenced with increasing complexity and diversity, misconception and mistakes are revealed. Students' new knowledge is error prone and needs to be fine tuned, reshaped, assimilated or better articulated with old knowledge. At the tuning stage, peer collaboration or discussion is adopted with social context structured by prototypes such as responsibility sharing, one working one critiquing, etc. This distribution of thinking loading is a form of scaffolding. But students' knowledge seldom overlaps completely, when conflict occurs, they are forced to examine their thinking, diagnose and evaluate problems hinted by the conflict, look for alternative perspectives, justify their perspectives through explanation and elaboration, reconstruct the perspectives in a more elaborate form, and at the same time keep an eye out for possible relevancies. Thus, this process is also a form of reflection. Furthermore, the product of socially distributed knowledge interaction such as comparing, evaluating, justifying, and criticizing has become part of the students' individual knowledge. Furthermore, two points should be noted here. First, in collaboration, the success of solving a problem is considered as the success of each participated student, and thus develop attitude to be open, seek out help and input from others. Second, it relieves insecurity when students see others to struggle and realize that difficulties are not unique to them, thus contributing to their confidence about self relative to others.

When children walk a few steps, mothers encourage them, perhaps with an explicit reward, to walk a few steps more. Besides increasing degree of mastery of the tasks, routinization or automatization is trying to solidify the acquired knowledge, better relate it to the conditions or situations where the knowledge is applicable, be able to predict, and bring students' performance close to expert performance. In other words, it helps to establish ownership of that knowledge or prevents it to become an 'inert' knowledge [Whit 16], an encapsulated information rarely to be accessed again. However, learning tasks at this stage are not so challenging as before and incentive is needed. A score board of achievement may be set up or student competition is employed to introduce social pressure. But effort, instead of ability, should be emphasized at that stage. From time to time, retrospection in the form of abstract reformulation of what has learned should also be taken at that stage.

As can be seen in Table 1, different stages of OCTR bring together most useful existing teaching strategies and place these strategies in a social learning repertoire. When comparing the OCTR model with the traditional classroom teaching, we shall find that most teachers only apply didactic description, story telling, and modeling, leaving the rest disregarded. On the other hand, while CAI community deals mainly with orientation and routinization (without social context), ITS community concentrates only on coaching.

Stages	**Objective**	**Strategy**
Orientation	Prior Knowledge Connection	Story Telling, Discovery Game,...etc.
Coaching	Knowledge Growth	Modeling, Scaffolding & Fading,...etc.
Tuning	Knowledge Articulation	Collaboration, Discussion,...etc.
Routinization	Knowledge Solidification	Competition, External Reward,...etc.

Table 1 Summary of OCTR Model

From the system design point of view, OCTR has two implications. Firstly, it serves as a conceptual framework that can be recurrently applied to designing a computer-based social learning system. For example, to teach the language LISP, we may organize the material according to OCTR, that is, O is for teaching concept, C is for helping student to solve some basic problems, and so on. When come to teach the concept, OCTR can be re-applied again. Of course, not every stage of OCTR has the same emphasis. Secondly, students' beliefs about self at the tuning and routinization stages are their perception in comparing with others. Besides the actual cognitive performance, this relative perception which determines their subsequent motivational state have to be modeled for guiding the system to behave.

5. A Quantitative Explanation of OCTR Model

In this section, we describe a quantitative formula which represents a simple relationship of some quantities and provides an interpretation of OCTR model. *Cognitive loading* (CL) defined below can be viewed as an approximation measure of individual learning ability at the stages of coaching, tuning, and routinization.

$$\text{Cognitive Loading} = \frac{(\Delta D\text{-}S)}{D*M} + M_p$$

Here D stands for the background domain knowledge or prior knowledge, S for the scaffolding offered by the system, M for the motivation of the student, Mp for the peer pressure which is regarded as part of motivation, and DD has two interpretations: knowledge increment and the degree of its assimilation with old knowledge.

Whether the formula can be used for monitoring the behavior of the system depends on the identification of a detailed set of learning variables contributing to the quantities comprising CL artificial intelligence technology to detect and measure the values of such learning variables. But here we are only interested in the explanation of OCTR model offered by this formulation. Suppose that the maxima of the cognitive loading that can be attained by an individual is a constant, then the overall objective of the system is, at any time, to maintain the detected CL as close to the student's maxima as possible.

At all stages, we intend to keep students highly motivated, that is, M is as large as possible. At the beginning, that is, during the orientation stage, we try to increase D, that is, to relate or activate as much the student's background knowledge as possible. This strategy is based on Kolodner's observation [Kolo 83] that a domain expert learns faster than a novice does. Thus, one can learn more effectively if one has a stronger background knowledge. Of course, if we can do that successfully, the attention and curiosity which are parts of motivation will also be increased. Note that, from the formula, if D and M are large, DD can be considerably increased. Thus, orientation is to enhance CL or to prepare for the next three stages.

The purpose of coaching is to let students learn to tackle a complete task. But, according to Martin's law [Wins 92], one cannot learn anything unless one almost knows it already. This means that teaching has to be in small steps. However, if the steps are too small, we hardly teach any new concept or task. The problem is how to maintain DD, the knowledge increment, at a level where the student can learn without degenerating intrinsic motivation from the task. However, at the beginning of that stage, DD is too large for the student to handle alone, thus, an appropriate amount of scaffolding, S, should be exerted to reduce CL. For modeling, we can view it as the case where S is very close to DD. Fading begins as coaching proceeds, which means that S decreases.

Next, at the tuning stage, the degree of assimilation or articulation of the new knowledge with the background knowledge is strengthened, that is, DD should be a robust part of D to facilitate future learning. Peer discussion or collaboration is utilized at this stage. On one hand, when DD is regarded as the degree of assimilation, DD can be measured as the intensity of emerging and vanishing issues when students are explaining, evaluating, and justifying each other's perspectives in resolving their cognitive conflicts; on the other hand, when DD is regarded as the knowledge increment, DD is no longer as large or critical as that in the coaching stage, yet, peer collaboration supplies a form of scaffolding, S. However, to keep the argument going, each student has to think and contribute a piece of the discourse. This mutual responsibility and expectation induce peer pressure. That is, when the first term of CL lessens, the second term, M_p, emerges.

In the routinization stage, the emphasis is to solidify the learned knowledge so as to ensure its accessibility or usability in the future. Since both DD and S have declined at this stage, the first term of CL vanishes. To maintain the momentum to learn, M_p, is marching to become a dominant term. In other words, students are engaged in a competitive environment where their innate impulse for survival drives them to face the challenge.

6. Development of LISA[2]

LISA is divided into seven sub-projects where each language is a sub-project and there is a beginning session on introduction of computer including computer history, architecture, etc. These sub-projects are currently being implemented in parallel for the orientation stage. Four faculties of computer science background, including the author, from National Central University, two cognitive psychologists from two other universities in Taiwan, and ten graduate students are involved in the project. We estimate that the complete courseware will take 70 hours for a reasonable student, covering the four stages of OCTR model.

In order to develop systems for students to use for today, not for next decade, we must treat students, teachers, and system administrator as colleagues. We do not simply deliver computer programs to users and then evaluate them. Students and teachers take part in the design process from the very beginning. Simple systems are produced initially, development continues to evolve in the course of recurrent periods of test, observation, reflection, and redesign until the system is able to actually share part of teacher's burden.

Apart from the learning companion, one-to-one, and group-to-group prototypes mentioned above, some sub-projects adopt some social learning prototypes and study how they can realize OCTR model [LiCC 92]. We describe briefly three of them here. The *synchronized distribution prototype* is a learning environment where all students will face the same computer teacher. If the teacher asks all students the same question, students will then communicate with their own computer companion before issuing an answer. Apart from having feedback from the teacher, statistics of the result of the students' answers will also be shown. Because all students learn from the same teacher, some synchronized mechanisms must be exercised during the learning process. This prototype is based on the anticipation of the integration of ISDN technology and satellite communication and schools may not be located at some centralized places.

The *small society dialog prototype* (Figure 1a) is a direct extension of the original learning companion prototype where instead of a three-agent learning environment, an additional computer student is added, forming a more complex four-agent prototype. The

[2] LISA is currently funded by National Science Council, Taiwan.

different levels of knowledge and persona of the two simulated students generate interesting issues that are not shared by others.

The *glassroom society dialog prototype* (Figure 1b), on the other hand, consists of two parts: the inner world (glassroom) and the outer world. The outer world is the small society dialog prototype mentioned above. The inner world is the subject to be observed and discussed by the outer world. The glassroom is a general term. It may be an object, an episode of video, or another small society. An example of this prototype in daily life perhaps is the operating room of a hospital where medical students outside the room observe the operation process through a glass window and is explained by an instructor some critical points in the process.

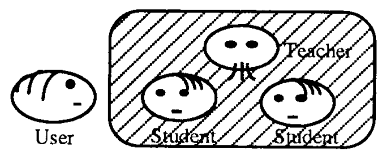

Figure 1a Small Society Dialog

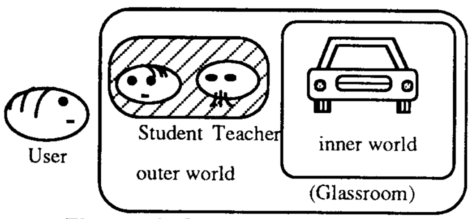

Figure 1b Glassroom Society Dialog

7. Conclusion

In this paper, we have presented the design framework of an on-going research project LISA as well as our initial results on its implementation. The psychological basis of our work is that individual learning is largely the re-enactment by the individual of cognitive processes that were originally experienced in society. The goal of the project is to explore some learning environments that may bring important impact to the future education. Related future work includes the critical problem of modeling student's social cognitive behavior, formulation of social dialogs in the system in learning, further investigation of knowledge-based architecture for intelligent multimedia CAI authoring systems, and designing visual language version of the six languages so that the potential population of users can be enlarged.

It is our hope that the project would serve to stimulate further research interest and development effort in building effective learning environments. In fact, of all the computer applications that are possible today, education seems to be the most socially significant. As the price of the computer is falling, the power of computer is increasing, and the technology of multimedia and network support is becoming more accessible, we have an opportunity to create new breeds of learning environments and, perhaps, to launch a revolution in education.

Reference

[AbSu 85] Abelson, H., and Sussman, G.H. Structure and Interpretation of Computer Programs. MIT Press, Cambridge, Mass. With J. Sussman, 1985.

[Bloo 84] Bloom, B. S. The 2 sigma problem: the search for methods of group instruction as effective as one-to-one. Educational Research, vol. 13, pp.4-16, 1984.

[Ch+ 92] Chan, T. W., Chung, Y. L., Ho, R. G., Hou, W.J. & Lin, G. L. "Distributed Learning Companion Systems -- WEST Revisited." The 2nd International Conference of Intelligent Tutoring Systems, Lecture Notes in Computer Science, Springer-Verlag, 1992.

[Chan 91] Chan, T. W. Integration-Kid: A Learning Companion System. The 12th International Joint Conference on Artificial Intelligence, Sydney, Australia, Morgan Kaufmann Publishers, Inc., pp. 1094-1099, 1991.

[Chan 92] Chan, T. W. Curriculum Tree: A Knowledge-Based Architecture for Intelligent Tutoring Systems, The 2nd International Conference of Intelligent Tutoring Systems, Lecture Notes in Computer Science, Springer-Verlag, 1992.

[ChBa 88] Chan, T. W. & Baskin, A. B. Studying with the Prince: The computer as a learning Companion, International Conference of Intelligent Tutoring Systems, 1988, June, Montreal, Canada, pp.194-200, 1988.

[ChBa 90] Chan, T. W. & Baskin, A. B. Learning Companion Systems. In C. Frasser & G. Gauthier (Eds.) Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education, Chapter 1, New Jersey: Ablex Publishing Corporation, 1990.

[ChCH 92] Chan, T. W., Chung, Y. L. & Hue, C. W. Evaluation of Distributed Learning Companion System, East-West Conference on Emerging Computer Technologies in Education, Moscow, 1992.

[ChWa 92] Chan, T. W. & Wang W. C. "Problem-Solving Model Approach for Teaching Multiple Introductory Programming Languages, East-West Conference on Emerging Computer Technologies in Education, Moscow, 1992.

[Clan 86] Clancey, W. J. Qualitative student models. In Traub, J. F. (Ed.), Annual Reviews of Computer Science, Vol. 1, pp. 381-450. Palo Alto, CA: Annual Reviews, Inc., 1986.

[CoBN 89] Collins, A., Brown, J. S., & Newman, S. E. Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser*, Hillsdale, NJ: Lawrence Erlbaum Associates Publishers, 1989.

[CuSe 90] Cumming, G. & Self, J. Collaborative Intelligent Educational Systems. In D. Bierman, J. Breuker, & J. Sandberg (Eds.), Artificial Intelligence and Education, pp. 73-80, Amsterdam: IOS, 1990.

[DiSe 91] Dillenbourg, P & Self, J. Designing Human-Computer Collaborative Learning. In C. O'Malley (Ed.), Human-Computer Collaborative Learning, Springer-Verlag, 1991.

[DiSe 92] Dillenbourg, P & Self, J. People Power: A Human-Computer Collaborative Learning System. The 2nd International Conference of Intelligent Tutoring Systems, Lecture Notes in Computer Science, Springer-Verlag, 1992.

[GiSe 88] Gilmore, D. & Self, J. The application of machine learning to intelligent tutoring systems. In Self, J. (Ed.) Artificial Intelligence and Human Learning, Intelligent computer-assisted instruction, pp. 179-196, New York: Chapman and Hall, 1988.

[JoJo 74] Johnson, D. W., & Johnson, R. T. Conflict in the classroom: Controversy and learning. Review of Educational Research, 49, 51-70, 1974.

[JoJo 75] Johnson, D. W., & Johnson, R. T. Learning together and alone. Englewood Cliffs, NJ: Prectice-Hall, 1975.

[Kolo 83] Kolodner, J. L. Maintaining organization in a dynamic long-term memory. Cognitive Science 7, p. 243-280, 1983.

[LiAS 77] Liskov, B., Atkinson, R., & Schaffert, C. Abstraction mechanisms in CLU. *Communications of the ACM* 20(8): p. 564-576, 1977.

[Lieb 86] Lieberman, H. Using prototypical object to implement shared behavior in object-oriented system. *Proceedingsof the First ACM Conference on Object-Oriented Programming System, Languages, and Applications*, 1986.

[PaSe 88] Parkes A. P. & John A. S. Video-based intelligent tutoring of procedural skills, International Conference of Intelligent Tutoring Systems, June, Montreal, Canada, 1988.

[Piag 67] Piaget, J. *Biology and Knowledge*. Paris: Gallimard, 1967.

[Self 86] Self, J. A perspective on intelligent computer-assisted learning. Journal of Computer Assisted Learning, Vol. 1, pp. 159-166, 1985.

[Self 86] Self, J. The application of machine learning to student modelling. Instructional Science, Vol. 14, pp. 327-338, 1986

[Slav 83] Slavin, R. E. Cooperative learing. New York: Longman.

[Sold 92a] Soldato, T. D. Motivational Planning. Brusilovsky, P. & Stefanuk, V., eds. Proceedings of the East-West conference on Emerging Computer Technologies in Education International centre for Scientific and technical Information. Moscow, Russia, 1992.

[Sold 92b] Soldato, T. D. Detecting and Reaction to the learner's motivational State. The 2nd International Conference of Intelligent Tutoring Systems, Lecture Notes in Computer Science, Springer-Verlag, 1992.

[Vygo 78] Vygotsky, L. S. *Mind in society: The development of higher psychological processes* (Cole, M., John-Steiner, V., Scribner, S., & Souberman, E. Eds.). Cambridge, MA: Harvard University Press, 1978.

[Whit 16] Whitehead, A. N. Address to the British Mathematial Society. Manchester, England, 1916.

[Wins 92] Winston, P. H. *Artificial Intelligence* , 3rd Edition. Addison Wesley, 1992.