

domain of learning how to interact with civilians in other cultures. For example, the *Tactical Language Training System (TLTS)* employed theory of mind intelligent agents in an educational environment focused on users practicing interactions with civilians in another culture using their native language (Johnson, Marsella, & Vilhjalmsson, 2004). An intelligent tutor would attempt to address motivational concerns based on techniques observed from real human language tutors. The *ELECT BiLAT* prototype (Hill et al., 2006) trains individuals on how to properly engage in bilateral negotiations within a cultural context. This system also focuses on virtual human non-verbal responses and dialogue. A virtual coach runs in the background and assess user actions, determining if the selected action contributed to a learning goal, whether or not to give explicit feedback, and gives targeted feedback based on a rudimentary student model. While these examples do not focus on health communication, they do provide useful examples of the current approaches to pedagogical conversational agents, whether they be agent you speak to directly or assistive agents that help via tutoring during conversations with agents.

Architecture Overview

This section provides an overview of the system design and details regarding each of the individual components. Throughout this section we provide justification for our design choices considering the requirements and constraints.

Figure 1 gives a pictorial view of the architecture of our system. As visible in the figure, the system is designed using one of the more common Web Application frameworks, the Model-View-Controller (MVC) framework. The justification for having a web framework was to make it available to users on the run (e.g. medical residents) without the need for an installation of any kind. The high-level flow for the system is as follows. The web page presents the user with contextual dialog choices that represent the steps in the BNI. The user selects one of these choices and that selection is forwarded to the servlet. The servlet calls the bean with this data, which processes this dialog and forwards the responses to the servlet that is eventually presented to the user. Below we describe the details of the view, the model and the database.

The View

The *view* is the face of the application, and has a dual purpose like in any MVC. It is responsible for taking the input from the user and passing it on to the controller and must also get the output from the controller and present the contents to the user. One of the constraints while designing for the view was the instructional nature of the software and the length of the BNI. Taking these into account, it was really crucial to keep the user engaged and motivated. For this reason we decided to have a life-like

patient and coach, to be designed in Flash. The Flex framework was an obvious fit considering the need for a Flash-based front-end and its infrastructure for support of Rich Internet applications.

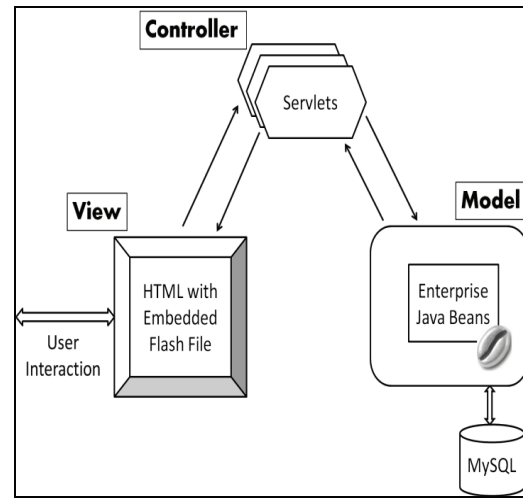


Figure 1. Overview of the Virtual BNI Trainer architecture.

The Model

The *model* is where the main cognition for the virtual coach resides. The model has been implemented as an Enterprise Java Bean. This bean implements both the coach and the patient model for this system. The reason for having an Enterprise Java Bean for a model was to be able to store user data across multiple requests. Although, we could have accomplished this using web sessions, we chose to use the model to store the session data. This choice was contingent upon the fact that we were dealing with a lot of session data that was needed to implement the user (Student) model described below.

The *controller* forms the input and calls the appropriate bean method with this input. The bean in turn uses the student, coach and patient models to retrieve the relevant information from the database and passes it on to the controller which then formats the output and sends it to the view. Details regarding the student, coach and patient models will be discussed in the next main section.

The Database

Implemented using MySQL, the database stores the entire authored dialog for the system along with user related data. While authoring the dialog, it was necessary for us to move away from simply taking the user through the BNI steps and introduce some complex dialog choices to fully gauge the understanding of the user. The database is responsible for storing all this in a structured format so that it can be retrieved quickly as response time is also of vital importance to the system.

It was important to establish metrics to understand the progress of the student to implement an intelligent and

adaptive student model. It is also crucial to be able to use past metrics data to provide the user with the most appropriate challenges that would allow him to improve his learning of the BNI. The VBT keeps track of the user progress and thus this information also persists in the database to provide this desired functionality

The Virtual Coach, Dr. Vicky

Our current approach to computer assisted BNI training centers on the use of a virtual coach, Dr. Vicky (see), who updates a model of student performance over time, provides feedback and remediation during training, and informs an after action review for users. This section discusses the decisions made in student model design,

Student Model

Having an adaptable student model is a common key component of intelligent tutoring systems. As explained in (Zhou & Evens, 1999), there exist various types of student model to address different issues. We are targeting student models for designing a customized solution path, responding with content appropriate feedback and also to assess the student level of mastery.

Several factors that influence student performance, such as ambiguous errors, skill deficiencies, cognitive slips, and random guesses can be taken into consideration while

constructing a student model (Katz, Lesgold, Eggen, & Gordin, 1994). Model-tracing methods are one such strategy, which includes an internal mapping of production rules students are expected to master (Koedinger, Anderson, Hadley, & Mark, 1997). By dividing the concepts into a series of topics in which student responses are monitored, it is possible to accurately access the student's current skill level when compared to the ideal system model. However, the rigidity of this system eliminates the ability to learn by "trial and error" (Katz et al., 1994). Other, less precise methods evaluate students based on their problem-solving strategies instead of specific responses. Our student model employs fuzzy set theory, which provides more flexibility in item categorization. Instead of categorizing skills into exclusive categories, it recognizes overlapping skills and captures the degree of overlap. We apply this fuzzy set theory to distinct knowledge variables using the methods used in the seminal work on SHERLOCK (Katz et al., 1994). The value in the set is increased or decreased depending on the level of evidence supporting the action. These variables, called *fuzzy variables*, are a set of attributes that represent a student's level of competence in a particular skill. We track five knowledge states for each of these variables, which include: no knowledge, limited knowledge, unautomated knowledge, partially automated knowledge, and fully developed knowledge.

The distinct knowledge variables that we track for each user stem from the requirement of having the users

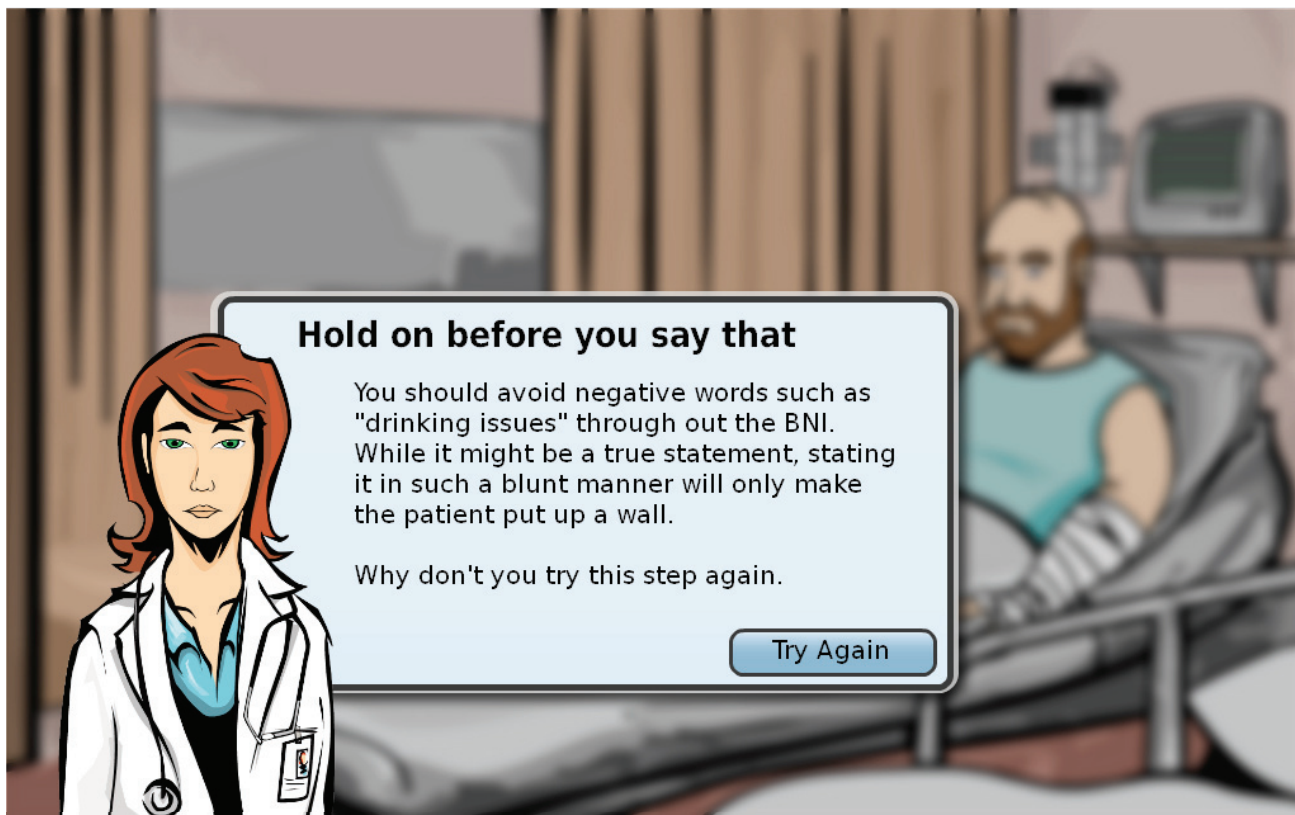


Figure 2. Screenshot of Dr. Vicky offering guidance to a user of the Virtual BNI Trainer (VBT).

User Errors

Dialogue is annotated with a categorization of correct, semi-correct, or fatal. Dialogue marked as correct contains no errors and is representative of how a BNI should be conducted. Semi-correct dialogue is mostly correct but contains an error or two. Selecting a semi-correct dialogue will allow the student to proceed but will make the virtual coach appear giving feedback on the error. Fatal dialogue is completely wrong or contains fatal words (e.g. “drinking problem”, “alcoholic”, or “drunk”) and will prompt the student to repeat the step. Every dialogue choice affects the student models; raising the competencies that were done correctly and lowering the competencies that were not.

interface needed to be inviting and easy to use both for initial users and long-term users. A quality backend architecture can be wasted if the frontend interface is not designed to match the user's needs. The VBT places a large focus on the environment, characters, and art design in order to provide an engaging experience. The menu structure is placed within the environment, floating in the screen, and not in the typical location of being docked to a corner or side.

A floating menu structure has the feel of fitting the created environment as opposed to being something added outside of the environment. Not only does this give a more modern feel, it also helps direct focus towards the avatars and give more weight to the environment they live in.



Figure 3. An example of the floating menu design used in the Virtual BNI Trainer.

Students can also make an ordering error by trying to execute a step in the BNI at the incorrect time. An example of an ordering error would be trying to negotiate a goal before establishing a problem. If the student selects dialogue in the wrong order the virtual coach will appear providing feedback and direct the user to try the step again. The student model will then be updated by deducing the order competency of the step the student is currently on.

User Interface Design

The goal of the user interface design was to provide a fresh modern look that promoted clear information display. The

The user has approximately nine dialog choices to choose from at any given time in the conversation. This could be confusing and time consuming to read if the dialog were all displayed at once so the VBT groups dialog choices into categories called buckets. When the student starts a new round of dialog in a conversation they are presented with approximately three buckets. Selecting a bucket expands it and displays approximately three dialog choices related to that bucket's category (three buckets times three dialog choices with in each equals nine total dialog choices). The exactly number of dialog and bucket choices depends on the difficulty level and location within the BNI process. Examples of bucket categories are "Greet", "Use Reflective

development work will address the authoring of more complex situations for intermediate and advanced users, as well as the development of mechanisms for advancing the user to the next difficulty. This should also include considerations for how to design patient attitudes, either through design or modeling, to generate less than ideal responses to proper BNI actions.

The second focus of our future work is on the individual differences that users have that may have a direct relevance to training situations. Learner motivation, mindset, goal orientation, and playstyle are just a few of the dimensions that shape how receptive learners are to educational digital media learning situations (Magerko, 2008). Our future work intends to explore these dimensions and their potential relevance to how the VBT can use not only models of the user at the knowledge, but across these broader dimensions to affect how and when guidance is given to the user and content is selected.

References

- Andre, E., Herzog, G., & Rist, T. (1988). On the simultaneous interpretation of real world image sequences and their natural language description: The system soccer. In *Proc. of the 8th ECAI* (pp. 449–454).
- Bickmore, T., & Gruber, A. (2010). Relational Agents in Clinical Psychiatry. *Harvard review of psychiatry*, 18(2), 119–130.
- Bickmore, T. W., Pfeifer, L. M., & Jack, B. W. (2009). Taking the time to care: empowering low health literacy hospital patients with virtual nurse agents. In *Proceedings of the 27th international conference on Human factors in computing systems*, 1265–1274.
- Bickmore, T., & Cassell, J. (2001). Relational agents: a model and implementation of building user trust. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, Seattle, WA, 396–403.
- Cassell, J., Bickmore, T., Billingham, M., Campbell, L., Chang, K., Vilhjálmsón, H., & Yan, H. (1999). Embodiment in conversational interfaces: Rea. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 520–527.
- D’Mello, S. K., Craig, S. D., Witherspoon, A., Mcdaniel, B., & Graesser, A. (2008). Automatic detection of learner’s affect from conversational cues. *User Modeling and User-Adapted Interaction*, 18(1), 45–80.
- D’Onofrio, G., & Degutis, L. C. (2010). Integrating Project ASSERT: A Screening, Intervention, and Referral to Treatment Program for Unhealthy Alcohol and Drug Use Into an Urban Emergency Department. *Academic Emergency Medicine*, 17(8), 903–911.
- Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H. H., Ventura, M., Olney, A., & Louwerse, M. M. (2004). AutoTutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers*, 36(2).
- Graesser, A. C., & Person, N. K. (1994). Question asking during tutoring. *American Educational Research Journal*, 31(1), 104–137.
- Green, N., Lawton, W., & Davis, B. (2004). An Assistive Conversation Skills Training System for Caregivers of Persons with Alzheimer's Disease. In *Dialogue Systems for Health Communication, Papers from the 2004 AAAI Fall Symposium, Technical Report FS-04-04*, AAAI Press, Menlo Park, CA (pp. 36–43).
- Hill, R. W., Belanich, J., Lane, H. C., Core, M. G., Dixon, M., Forbell, E., Kim, J., et al. (2006). Pedagogically structured game-based training: Development of the ELECT BiLAT simulation. In *Proceedings of the 25th Army Science conference*, 82–95.
- Johnson, W. L., Marsella, S., & Vilhjálmsón, H. (2004). The DARWARS tactical language training system. In *Proceedings of IITSEC*. Marina del Rey, CA.
- Kirschner, P. A., Sweller, J., & Clark, R. E. (2006). Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational psychologist*, 41(2), 75–86.
- Magerko, B. (2008). Adaptation in digital games. *IEEE Computer Magazine*, ed. van Lent, M., 41(6), 87–89.
- Marsella, S., Johnson, W. L., & LaBore, C. (2003). Interactive pedagogical drama for health interventions. In *Conference on Artificial Intelligence in Education, Sydney, Australia*.
- National Institute on Alcohol Abuse and Alcoholism. (2007). Helping Patients Who Drink Too Much: A Clinician's Guide. *Helping Patients Who Drink Too Much: A Clinician's Guide*. Retrieved October 18, 2010, from http://pubs.niaaa.nih.gov/publications/Practitioner/CliniciansGuide2005/clinicians_guide.htm
- Piwek, P., Krenn, B., Schröder, M., Grice, M., Baumann, S., & Pirker, H. (2004). RRL: A rich representation language for the description of agent behaviour in NECA. *AAMAS Workshop: Embodied Conversational agents - Let's Specify and Evaluate Them!*, Bologna, Italy.
- Zhou, Y., & Evens, M. W. (1999). A practical student model in an intelligent tutoring system. In *ICTAI*, 13–18.