Intelligent Tutoring System Using Hybrid Expert System With Speech Model in Neural Networks

R.Venkatesh, E.R.Naganathan, N.Uma Maheswari

***Abstract*—Many of the intelligent tutoring systems that have been developed during the last 20 years have proven to be quite successful, particularly in the domains of mathematics, science, and technology. They produce significant learning gains beyond classroom environments. They are capable of engaging most students' attention and interest for hours. This paper aims to establish some characteristics, properties and functions that an ITS should provide combined with speech, and the possible contributions that the different fields of research can make, proposing a multi-domain and multidisciplinary framework to address the research in this field. The framework incorporates a knowledge base where data and knowledge related to the problem are maintained and a model base related to student, teaching and environmental issues together with pedagogical perspectives. A theme underlying much of ITS research is domain independence, i.e. the degree to which knowledge encoded in the teaching model can be reused in different domains. Although to the external observer, domain independence seems like an essential characteristic of intelligence, many experts believe that some of the essential pedagogical knowledge in every domain is fundamentally domain-dependent. The proposed work was used for implementing ITS using supervised learning neural networks to a successful rate. Instead of being mere information-delivery systems, our systems help the students to actively construct knowledge.**

***Index Terms*—Domain Independence, Intelligent Tutoring utoring Systems(ITS), Neural Networks , Supervised Learning.**

I. Introduction

Intelligent Tutoring Systems (ITS) provide the benefits of one-to-one instruction in an automatic way and cost effectively, keeping in mind their multidisciplinary nature. The challenge remains on transporting to computers the expertise, skills and mode of action of the human tutor, overcoming space, time, socio economical and environmental restrictions. ITS appear as a form of deployment of this issue and have been object of an increasing research. One reason that ITS is such a large and varied field is that "Intelligent Tutoring System" is a broad term, encompassing any computer program that contains some intelligence and can be used in learning. ITS is an outgrowth of the earlier computer-aided instruction or CAI model, which usually refers to a frame-based system with hard-coded links, i.e. hypertext with an instructional purpose. The traditional ITS model contains four components: the domain model, the student model, the teaching model, speech model and a learning environment or user interface. ITS projects can vary tremendously according to the relative level of intelligence of the components. For example, a project focusing on intelligence in the domain model may generate solutions to complex and novel problems so that students can always have new problems to practice on, but it might only have simple methods for teaching those problems, while a system that concentrates on multiple or novel ways to teach a particular topic might find a less sophisticated representation of that content sufficient.

II. System Architecture for Intelligent tutoring system

The proposed basic architecture of the ITS consists of the following components:

1) -Domain knowledge containing meta description, knowledge concepts and course unit
2) -Pedagogical Model containing concept neurule, course unit neurules and evaluation module
3) -User modeling component containing models of the system's users and mechanisms for creating these models.
4) -Student model which encompasses
5) Question Answering system (student input, value status, derivation procedure and interface preferences)
6) -Speech model which contains Speech Act Classification(SAC) and task activity

The ITS is based on an expert system aiming to control the teaching process. In the following sections, we elaborate on the system's key aspects.

A. Domain Knowledge

The domain knowledge contains knowledge regarding the subject being taught as well as the actual teaching material. It consists of three parts:

(a) Meta Description
(b) The knowledge concepts
(c) The course units (variables, relationships).

The knowledge concepts refer to basic pieces of knowledge concerning the domain. A concept can have links to other concepts. These links denote its prerequisite concepts. In this way, one or more concept networks are formed representing the pedagogical units constitute the teaching material presented to the system users. Each course unit is associated with a number of knowledge concepts. A course unit may present theory, may be an example or an exercise.

To facilitate the selection and ordering of the course units, the domain knowledge includes a meta-description of the course units based on their general attributes. Main such attributes for a course unit are its level of difficulty, its pedagogical type (theory, example, exercise), its multimedia type, the required Internet connection, etc. The domain knowledge is aimed to store, manipulate and reason with knowledge of the domain being taught. A lot of work has been done with artificial intelligence techniques to model student's reasoning with Back propagation networks.

B. Pedagogical Model

The pedagogical model represents the teaching process. It provides the knowledge infrastructure in order to tailor the presentation of the teaching material according to the information contained in the user model. As shown in Figure 1, the pedagogical model consists of three main components:

1) Concept neurules,
2) Course units' neurules and
3) Evaluation module.

The task of the concept neurules is to construct a user adapted lesson plan by selecting and ordering the appropriate concepts. This is based on the user's knowledge of the concepts, the user's domain knowledge level, the concepts' level of difficulty and the links connecting the concepts. According to the plan constructed by the concept neurules, the course units' neurules select and order the course units that are suitable for presentation. For this purpose, the student characteristics of the user model as well as the metadescription of the course units are taken into account. The final evaluations provide us a model that allows prediction of student detail.

The evaluation module evaluates the user's performance and updates accordingly the user model. More specifically, it assigns knowledge values to the concepts based on the interaction parameters and updates the inferable student characteristics based on the classification neurules of the user modeling component. When the user gains an acceptable knowledge level of the concepts belonging to the initial lesson plan, a new plan is created. Intelligent Tutoring System(ITS) is a system that uses Feed forward Backpropagation and it has been trained with a group of student data to predict student results. Various tests have been conducted to examine adherence to real time data. We have predicted the Student Performance with the previous results and upcoming results

III. Evaluation Module for ITS(Value correction and procedure)

1. Inputs are used for training the Neural Networks and the Desired output is got from the Exam Results.
2. To implement the Backpropagation Network based on the inputs
    - Set the Number of input nodes in the Input layer
    - Set the Number of hidden nodes in the Hidden Layer
    - Set the Weight Value
3. Training is done in the Feed forward pass in the Backpropagation Algorithm
4. Testing is done in the BPN and the Output Vector was found.
5. Comparison was made between the Actual Output and NN Output.

IV. User Modeling Component

The user modeling component is used to record information concerning the user which is vital for the system's user-adapted operation. It contains models of the system's users and mechanisms for creating these models.

The user model consists of four types of items:

1) Personal data(e.g. name, email),
2) Interaction parameters,
3) Knowledge of the concepts and
4) Student characteristics.

The student characteristics and knowledge of the concepts directly affect the teaching process, whereas the interaction parameters indirectly. The Interaction parameters form the basis of the user model and constitute information recorded from the interaction with the system. They represents things like, the type and number of the units accessed, the type and the amount of help asked, the answers to the exercises etc.

The student characteristics include items such as the multimedia type preferences (e.g. text, images, or animations) regarding the presented course units, the domain knowledge level, the learning ability level, the available Internet connection etc. Based on the way they acquire their values, student characteristics are discerned into two groups, directly obtainable or inferable. The directly obtainable characteristics obtain their values directly from the user whereas the values of the inferable ones are inferred by the system based on the interaction parameters and the knowledge of the concepts.

A neurule base containing classification neurules is used to derive the values of the inferable characteristics. The user models are updated during the teaching process**.** The user's knowledge of the domain is represented as a combination of a stereotype and an overlay model. The stereotype denotes the domain knowledge level. The overlay model is based on the concepts associated with the course learning units. More specifically, each concept is associated with a value denoting the user knowledge level of this concept.

V. Student Model

It stores information specific to each individual learner. The minimum requirement of the student model would be to track the performance of the student. The student model evaluates each learner's performance to determine his or her knowledge, perceptual abilities, and reasoning skills by using Question Answering System. The student model contains their input for a particular question, the derivation procedures etc.,

Student A	32 +39 51	50 +37 73
Student B	42 +39 161	36 +37 183
Student C	20 +39 62	41 +37 85

Figure 2. ITS Student Modeling Example

A. Question Answering system

The Questions from a particular course is queried from the student and the answer is obtained using the Question Answering System. The keywords are identified from the asked question and it is matched with the text stored in knowledge base using neural network and the corresponding output is displayed. The primary focus in designing a Question and Answering system for ITS is that the answer that is given for a particular question should be precise and to the point irrelevant answers can mislead a student and dissuade a student from using the system.

Artificial neural networks resemble the human brain in the following two ways:

1) An ANN acquires knowledge through learning.
2) An ANN's knowledge is stored within inter-neuron connection strengths known as synaptic weights.

The advantage of ANNs is their ability to represent both linear and non-linear relationships and in their ability to learn these relationships directly from the historical data. Traditional linear models are simply inadequate when it comes to modeling data that contains non-linear characteristics.

VI. Neural Model

Neural Model is used for Question Answering system. The questions on the wanted subject are put into the neural model. Both the possible Questions and Answers are stored in the desired Output. Now the questions that are given as input to the ITS are trained with questions in the desired and then their corresponding answers are displayed to the students.

By applying data from sub files to input vector, the net input was calculated by using adder function,(i.e.) net value. The net is processed by an activation function to produce the neurons to give output.

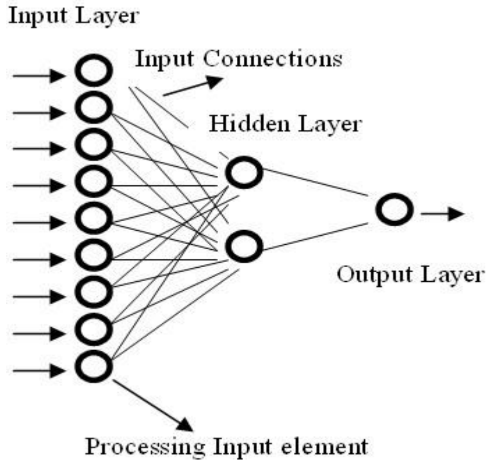

Figure 3. NN System Architecture

Here, F is called a squashing function. Here the model uses sigmoid function (meaning S –shaped),

$$F(net) = out= 1/1+e\text{-}net \quad (1)$$

then the desired output is subtracted from the actual output to find the error and it is propagated backwards to minimize the error by adjusting weights.

VII. Speech Model

For maintaining and referring to a detailed model of each user's strengths and weaknesses, the ITS can use dialog system which makes ITDS that can provide highly specific, relevant instruction. In more complex domains, the tutoring system can monitor a learner's sequence of actions to infer his or her understanding. This system applies pattern-matching rules to detect sequences of actions that indicate whether the student does or doesn't understand.

A. Language Extraction

A tutor segments words and punctuation marks with 99% accuracy, assigns alternative syntactic tags to words with 98% accuracy, and assigns the correct syntactic class to words (based on context) with 93% accuracy. Each word the learner asks during a conversational turn is analyzed in terms of its alternative syntactic classes, called part-of speech tags in the linguistic community. A total of 17 alternative syntactic tags are used. These include classes such as noun, pronoun, verb, auxiliary verb, adjective, adverb, conjunction, preposition, question words, digits, and operators. Because many words may be categorized into more than one syntactic class, a neural network is used to assign the correct syntactic class. This is done by taking into consideration the syntactic class of the preceding word and the class of the subsequent word.

B. Speech Act Classifier

The tutor uses strings of words and punctuation to classify each contribution of the learner into speech act units. A neural network is 96% accurate in classifying each speech segment into one of 46 phoneme categories using Recurrent Neural Network.

Then the text corresponding to the input was displayed by the learner using Hidden Markov Model form the language extraction method. A tutor has different strategies for responding to the various categories of speech acts, which also vary with the quality of the learner's contribution. When the learner's contribution is a definition question, then the question is matched to entries in the stored database and the tutor produces the definition if there is a high match. The various methods of handling different speech acts are needed to insure smooth mixed initiative dialog, while attempting to bring out what the student know. The tutor needs to determine the intent or conversational function of a learner's contribution in order to respond to the student flexibility when the learner asks a question. Extensive testing of the classifier showed that the accuracy of the classifier ranged from 75% to 97%, depending on the speech corpus.

C. File Management

The file management is used as a marker for the lecturer's mode answer which can be Communicated with the ITS module.

D. Modes

The modes select the person to be communicated with the ITS whether a student, a lecturer or from the administration.

VIII. IMPLEMENTATION

Our experimental procedure, taking roughly 3 hours/student, is as follows: students 1) read a small document of background material, 2) take a pretest measuring their physics knowledge, 3) use ITS Dialogue to work through 5 mathematical problems, and 4) take a post-test similar to the pretest. We have collected 50 dialogues from 10 students (12 total hours of speech, mean dialogue time of 30 minutes). ITS Dialogue uses 30 dialogue-state dependent language models for speech recognition; 23 of these 16 models have been used to process the data collected to date. These stochastic language models were initially trained using 1000 typed student utterances, then later enhanced with spoken utterances obtained during ITS Dialogue's testing. For the 1600 student turns that we have collected, ITS Dialogue's current Word Error Rate is 20%. While this is the traditional method of evaluating speech recognition, semantic rather than transcription accuracy is more useful for dialogue evaluation as it does not penalize for word errors that are unimportant to overall utterance interpretation. Semantic analysis based on speech recognition is the same as based on perfect transcription 90% of the time.

IX. CONCLUSION

The models developed in this work enable teaching systems to be developed in various fields and subjects. The ITDS allows tutored instruction systems to be developed independently on any subject. Intelligent tutors generated are easy to modify: all that is required is to include the name of the new area (with the associated pedagogical information) in the list of sub areas. The area introduced will appear the next time the system is run. It is quite easy for non-computing personnel to develop auto-regulated intelligent tutors, as they only have to fill out record cards and lists for which almost no knowledge of programming is required. This last point only concerns to programming tasks because ITDS pedagogical strategies are very difficult to set up. The tool assists the development of tutors by checking, in development time, that the database is complete and that no compulsory object or property is left undefined. If this happens, the system tells the developer to complete the required information. The time and cost required for developing auto regulated intelligent tutors are dramatically reduced. The improvement entailed by the tool not only reduces development times but also appreciably simplifies the technical knowledge required of personnel involved in the generation of an auto-regulated intelligent tutoring dialogue system. In fact, the developers of new ITDS have not expert programmers and do not have much technical knowledge

REFERENCES

[1] Aist G, Kort B, Reilly R, Mostow J, and Picard R. Experimentally augmenting an intelligent tutoring system with human-supplied capabilities: Adding Human-Provided Emotional Scaffolding to an Automated Reading Tutor that Listens. In Proc. of Intelligent Tutoring Systems, 2002.
[2] Aleven V and Rose C. P. 2003. Proc. of the AIED 2003 Workshop on Tutorial Dialogue Systems: With a View toward the Classroom.
[3] Aleven V, Popescu O, and Koedinger K 2001. Towards tutorial dialog to support selfexplanation Adding natural language understanding to a cognitive tutor. In J. D. Moore, C.L. Red_eld, and W. L. Johnson, editors, Proc. of Arti_cial Intelligence in Education, pages 246.255.
[4] Ang J, Dhillon R,. Krupski A, Shriberg E, and Stolcke A. 2002. Prosody-based automatic detection of annoyance and frustration in human-computer dialog. In Proc. Of ICSLP.
[5] Batliner A,Fischer K, Huber R, Spilker J, and Noth E ,2003. How to trouble in communication. Speech Communication, 40:117.143.
[6] Chi M, De Leeuw N, Chiu, and Lavancher.1994. Eliciting self-explanations improves understanding. Cognitive Science, 18:439.477.
[7] Devillers, Lamel, and Vasilescu. 2003. Emotion detection in task-oriented spoken dialogs. In Proc. Of ICME.
[8] Michaeland, and Allen Rovick. 2001. Circsimtutor: An Intelligent Tutoring System Using Natural Language Dialogue. In Proc. Midwest AI and Cognitive Science Conference.
[9] Forbes-Riley and Litman. 2004. Predicting emotion in spoken dialogue from multiple knowledge sources. In Proc. Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics.
[10] Fry, Ginzton, Peters, Clark, and Pon-Barry. 2001. Automated tutoring dialogues for training in shipboard damage control. In Proc. SIGdialWorkshop on Discourse and Dialogue.
[11] Graesser, Person, and Harter et al. 2001. Teaching tactics and dialog in Autotutor International Journal of Artificial Intelligence in Education
[12] Hausmann and Chi. 2002. Can a computer interface support self-explaining? The International Journal of Cognitive Technology, 7(1).
[13] Jordan, Makatchev, and VanLehn. 2003. Abductive theorem proving for analyzing student explanations. In Proc. Artificial Intelligence in Education.
[14] Lee, Narayanan, and Pieraccini. 2002. Combining acoustic and language information for emotion recognition. In Proc. of ICSLP.

Venkatesh .R received his B.Sc in Computer Science from Madurai Kamaraj University in India , in 1991 and M.Sc. in Computer Science from Alagappa University in India, in 1994. He is a faculty member in the department of Information Technology in RVS College of Engineering & Technology, Dindigul, in India..His current research interests include artificial intelligence , Neural Network , soft computing.

Naganathan E.R received his Ph.D in Computer Applications from the Alagappa University, Karaikudi, India in 2000. He was a Assistant Professor in the Department of Computer Science, Jerash University, Jerash, Jordan from 2001 to 2003. Currently, He is a Profesor and Head of the Department in the Department of Computer Applications at the Velammal Engineering College, Chennai, India.

Umamaheswari .N received her M.E in Computer Science & Engineering from the Madras University, Chennai, India in 2002. Currently, She is a Assistant Professor in the Department of Computer Science & Engineering at the P.S.N.A. College of Engineering & Technology, Dindigul, India.His current research interests include Artificial Intelligence, Speech Processing, Neural Networks and Soft Computing

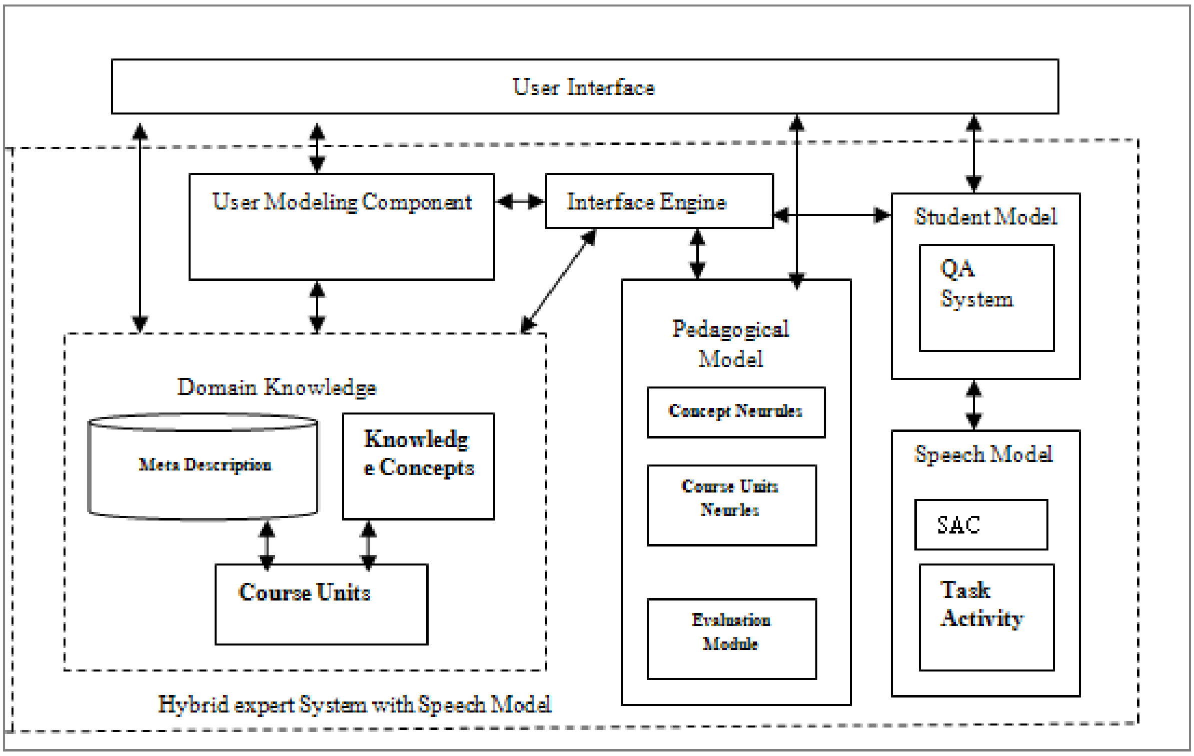

Figure 1. System Architecture for ITS