CogSysIII Lecture 10: Intelligent Tutor Systems

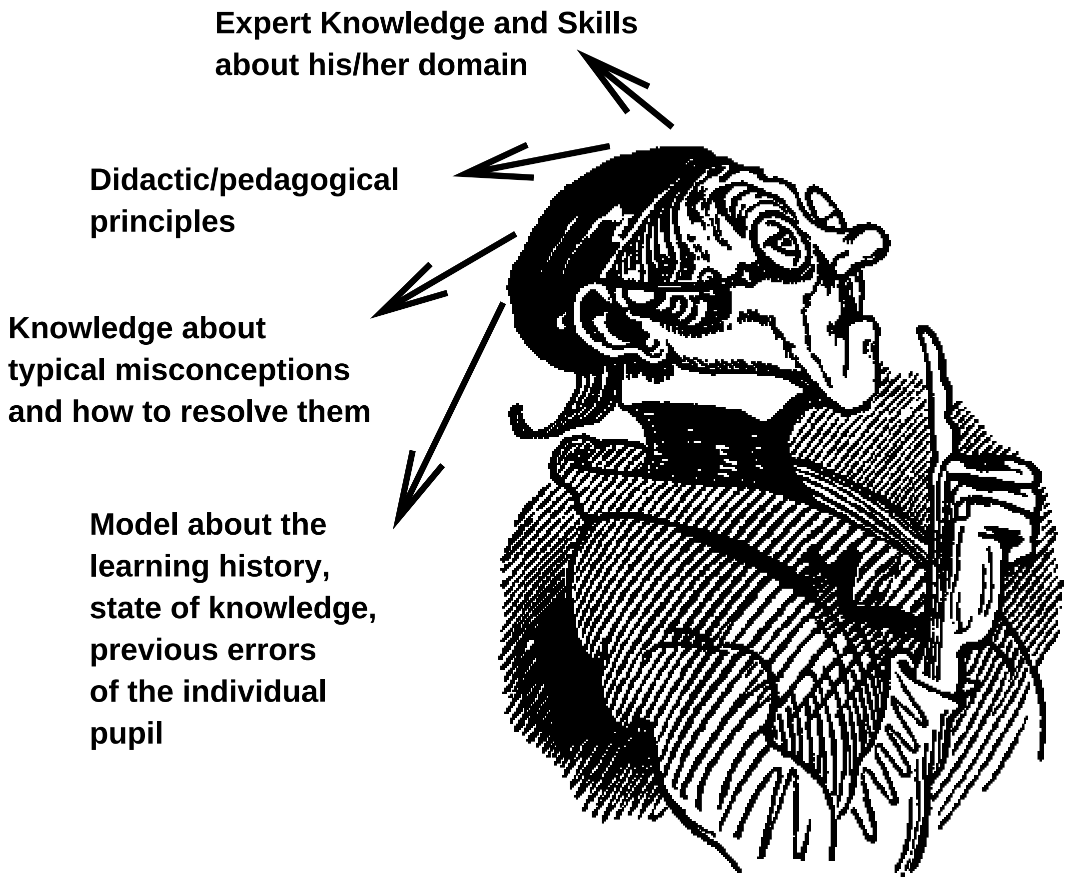
Human Computer Interaction

Ute Schmid

Applied Computer Science, University of Bamberg

last change July 3, 2007

Intelligent Tutor Systems



- see lecture of Frank Puppe, Würzburg:
  http://ki.informatik.uni-wuerzburg.de/teach/ss-2003/its/uebungen/
  Introduction (1), ITS (4), History of ITS (5)

Learning

- Change of behavior caused by experience
  - Declarative: Extension of knowledge
  - Procedural: Skill acquisition
- Improvement by restructuring or incorporation of new knowledge/skills
  - Faster or more accurate performance (power law of learning)
  - New behavior

Types of Learning

- Rote Learning
  The capital of France is Paris.
  5 times 7 is 35.
  A dog is a mammal.
- Concept Learning
  Dog, Ancestor, Preparation to an exam such that I pass
- Skill and Strategy Acquistion
  Making a proof by induction, preparing a travel
- Concept learning and Skill acquisition is learning from examples, that is, inductive learning! (see lecture CogSysII)

Paradigms for Learning

- Behaviorism
  - “Brain” as Black Box
  - Teaching: Connecting the stimulus with the “correct” response
  - Positive (or negative) Feedback
  - “Drill and Test”
- Cognitivism
  - Information processing paradigm
  - Teaching: support for problem solving, learning by doing
- Constructivism
  - Learners build personal interpretations of the world based on individual experiences and interactions.
  - Teaching: Align and design experiences for the learner so that authentic, relevant contexts can be experienced.

Learning with the Computer

- Independence of time/place
- Individual speed of learning, individual interests
- Individual strategies of learning (?)
- Relieve of teachers (costs, availability, quality ensurance)
- Learning dangerous activities (e.g. flight simulator)
- Learning in the context of realistic models
- Problems: social component of learning (model learning), restricted human-computer communication, trade-of of effort and utility, there are many bad systems

Didactic Principles

- Motivation: intrinsical vs. extrinsical
- Encouragement of creativity: give experience of perception with all sense, allow new ideas
- Differenciation: working with homgenous groups students
- Structuring: clear topic, separate in clear parts, systematic introduction of concepts and rules
- Illustration and exemplification
- Exercise and success control: allow for success (without making things trivial), relevance

Dimensions of Tutor Systems

- Tutor strategies
  - Explaining
  - Observing and Helping
  - Mentoring and Coaching
- Content: facts and context free rules; problem solving, complex situations, ...
- Goals: remembering, applying, understanding, discovering, ...

Types of Systems

- Presentation and Browsing
- Drill and Test
- Tutor Systems
- Intelligent Tutor Systems
- Simulations

Presentation and Browsing

- Restricted interaction, fixed branching
- Hypertext and Hypermedia
- Cheaper than books, full text search

Drill and Test

- Only tests, with statistics (percent correct answers etc.)
- Vocable trainer, medicine exams, driving licence
- Wer wird Millionär?
- Types of Questions
  - Closed: Multiple Choice, Rank order, cloze (Lückentext)
  - Open: input of numbers, text

Tutor Systems

- Teaching of new contents/skills, Question Answering by the system
- Simple tutor systems do *not* have an internal student model!
- Computer-aided instruction: module and questions, depending on the answers repetition with variations or progress
- Frame-oriented CAI: discrimination between error types and branching (variations) in dependence of the recognized type (predecessor to student model in ITS)

Simulation

- Technical or social simulation
- Can be based on causal mental models/qualitative reasoning
- Student actions are executed in the simulation model, this is the only form of feedback
- Dynamical, high degree of interaction
- E.g. simulation systems for complex problem solving (Dörner)
- E.g. SOPHIE (early system of the 70ies): debugging of electronical circuits
- E.g. LOGO (Papert, late 60ies)
- compare to Lego, electronical kits

ITS Overview

- Simple tutor systems no nothing about what they teach and whom they teach
- 70ies: with the beginning of AI reasearch, extension of tutor systems with internal knowledge modules
- Paradigm: cognitivism
- System has its own inference component
- Because of internal models: generation of interaction sequences, exercises; system is able to answer questions

Overview Cont.

- Knowledge models:
  - domain model
    e.g. SOPHIE (geography of south america), allows mixed initiative dialogs
  - learner model
    e.g. GUIDON (tutor extension of the expert system MYCIN), allows diagnosis of didactical relevant states of the student
    different data sources (dialog history, behavior of the student, goal-directed intervention)

Components of ITSs

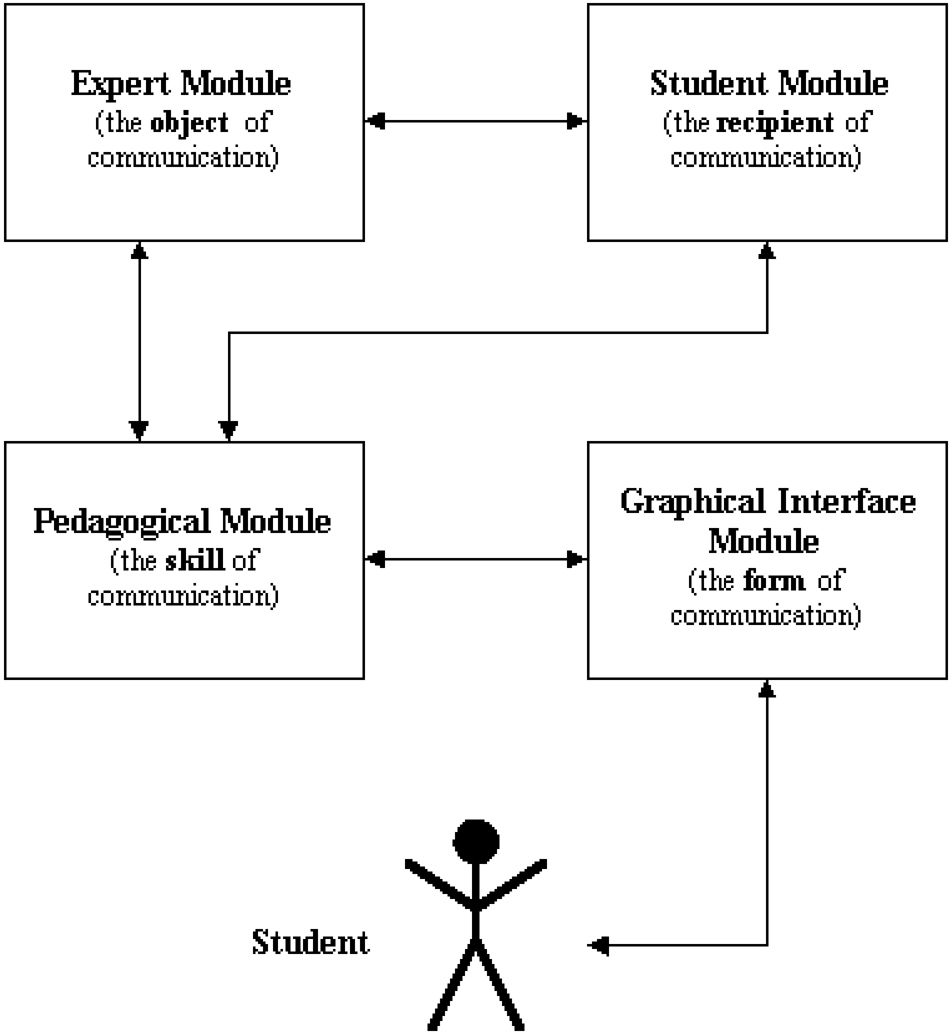


Figure 6.2 Anatomy of an Intelligent Tutoring System

Source:
http://home2.pacific.net.s
~ auddrick/its.html

Components of ITSs

- GUI: often involves the most effort: robust and intuitive
- Didactic component: lecture modules, plan of sessions, pedagocial knowledge, pedagocical strategies
- Student model: model of the knowledgte of the student, base for planning further didactical procedure
- Domain model: expert knowledge and skill, often executable, inference component allows to answer student questions

Applications of ITSs

- Medicin (GUIDON): problem based learning with generated feedback
- Mathematics (e.g. Geometry Tutor based on ACT): learning algebra or proofs in geometry with step-by-step application and generated feedback
- Programming (e.g. Lisp Tutor based on ACT)
- many more: see history of ITS

Pros and Cons of ITSs

- Pro: Highly flexible and adaptive
- Main Problem: very high effort, esp. for domain model and for some types of student models
- Restricted language for the student-tutor diaglogs
- No inter-human student-teacher relationship

Domain Model

- Source of Knowledge and standard for evaluation of student inputs
- must be communicable (simple, **symbolic** repres.)
- simple processing (no complex numerical calculations such as Bayes), no complex control structures (best is linear, no nesting of loops etc)
- Relation to previous knowledge of students
- May be knowledge on different competence levels
- Knowledge Represenation with: semantic nets, frames, rules, constraints, qualitative or quantitative simulation
- Diagnostic knowledge models: decision trees, heuristic diagnosis, case-based diagnosis
- Agent models for distributed knowledge, ...

Student Model

- What does the student know? (know that)
- What can the student do? (know how)
- What did the student do until now? (history)
- What type is the student?
- Input/Generation:
  - explicit: stereotype selection
  - implicit interrupting: interruption, explicit enquirey
  - implicit automatically: monitoring

Types of Student Models

- Stereotypes
- Error Libraries
- Overlay Models
- Extended Overlay Models
- Constructive Models

Stereotypes

- Facts, attributes and preferences which are interrelated
- Example intermediate student knows all simple knowledge units and some medium-level knowledge units

Error Libraries

- Collection of typical errors and misconceptions of students
- Select an item from the library during a tutoring situation
- Simple and flexible, but high costs in assesment

Overlay Model

- Domain model is extended by annotations: which knowledge unit is known by a student
- simple but not flexible
- implicit assumption: errors are due to **incomplete** knowledge
- Extended Overlay Model: additionally taking into account errors and misconceptions; can be generated from knowledge elements or from error library (very good but very expensive)

Dynamically Constructed Model

- Use machine learning to generate models about student knowledge
- highly flexible but difficult to obtain

Applications

- WHY, SOPHY: no student model
- Geography Tutor SCHOLAR: overlay model initiative dialog
- BUGGY (and others): error libraries
- LISP-Tutor: extended overlay (correct and erroneous production rules)

These and other classical systems are presented in lecture 5 of Frank Puppe.

Problems with Student Models

- How good can a student model be?
  - inconsistencies in behavior due to motivation, concentration, “luck”
  - Combinations of errors can produce unexpected (maybe even correct) behavior
- Clustering students in learning psychological stereotypes (cognitive styles; motivational styles) is controversial
- Privacy and data security

Didactical Module

- Which paradigm? which kind of learning? (reception, transfer, problem solving, ...) which competences? (cognitive, organizational, communicative ...)
- Initiative: system, learner, mixed
- Presentation: sequence, degree of guidance
- Tasks: number, time, complexity
- Feedback: eager or by request, hints, explanations, examples

Dealing with Student Errors

- What kind of error?
  - Profound error in content/in learning progress?
  - Error rates; repetition of same error?
  - Error recognized with certainty?
  - Reaction: ignore, protocoll, hint, insist on correction
- Typically realized by set of rules

Explanations and Evaluations

- Explanations by the student (socratical dialog): open or closed form inputs; evaluation by tutor system, relation to explanations of domain model
- Explanations of the system: pre-formulated or generated, general or situational, automatically or by request, degree of details
- Evaluation: generated or static, situational or dependent on result

GUI

- Robust, efficient, easy to learn and use
- Interaction language: commands, restricted natural, natural
- Interface to model: icons, menues, graphics
- Flexible navigation (event based vs guided)
- Taking account of context, learning history

requirements for GUIS see last slides in lecture 4 of Puppe

Current Trends

- Reuse and standardisation of content
- Support of collaborative learning, distant learning
- Adaptive tutor systems
  - e.g. Netcoach