

Status quo und Perspektiven der Lernprogrammentwicklung und des Lernprogrammeinsatzes

M. G. Zilahi-Szabo, Giessen

1. Die 60er Jahre/70er Jahre - Beginn, Euphorie

1.1 Chronologische Zeittafel

Von einem meßbaren und merklichen Einsatz des Computers zur Entwicklung und zur Nutzung von sog. Lehr- oder Lernprogrammen kann in der Bundesrepublik Deutschland erst seit Beginn der 70er Jahre, nach Inkrafttreten des 2. DV-Förderungsprogramms gesprochen werden (Abb. 1). Durch die damals bereitgestellten Finanzmittel und auf der Grundlage der in den USA bereits Ende der 60er Jahre gemachten Erfahrungen wurde eine Forschungsperiode eingeleitet, deren Inhalte und Resultate für die gegenwärtige Diskussion in bezug auf die Benutzer-Ergonomie, Software Engineering u. a. Teilgebiete der Anwendersoftware-Entwicklung schlechthin von eminenter Bedeutung sind. Ihre Auswirkungen waren sicherlich auch für weite Teile der angewandten Informatik von hohem Wert.

| | | |
|------------|---|---|
| Früher | : | Buch, Schallplatte, Kassette |
| | | |
| 60er Jahre | : | Programmierte Unterweisung (PU) |
| | | |
| 70er Jahre | : | Computerunterstützter Unterricht (CUU) |
| | | |
| 80er Jahre | : | Computer Based Training (CBT) |
| | | |
| Zukunft | : | Multi-(Hyper)media, Hypertext, neuronale Netze vernetzte Lernsysteme |

Abb. 1: Chronologische Entwicklungstafel

Hypertexte sind nicht aufeinander folgende verbundene Textstücke oder andere Informationen. Die Elemente, die verbunden werden können, heißen Knoten. Das ganze System wird von einem Netz von Knoten gebildet. Verbindungen können in zwei Richtungen angelegt sein. Der Leser erreicht "seine" Information in den Knoten, in dem er durch die Verbindungen navigiert wird. Die Navigation vermittelt einen Überblick über die Struktur der Informationen.

1.2 Einsatzgebiete des Computers in der Lehre

Der Einsatz des Computers in der Lehre hat sich in den vergangenen Jahren auf verschiedene Sachgebiete konzentriert, und zwar auf:

- Wissensvermittlung
- Wissensanwendung/Training
- Üben von Fertigkeiten

- Beratung
- Wissenschaftsüberprüfung
- Selbststudium
- Ablaufsteuerung in der Lehre
- Medien-Einsatz-Steuerung im Lehr- und im Lernprozeß.

Die im folgenden aufgeführten Bestimmungsfaktoren von Lernprogrammen sind Indikatoren, die anzeigen, unter welchen Bedingungen gewisse Phänomene sichtbar werden:

- PERSONELLE FAKTOREN
 - . Denkgewohnheit,
 - . Motivation, Aktivierung,
 - . Konfliktsituation, Individualisierung,
 - . Wissensstand, Schwierigkeitsgrad, Ausbildung,
 - . Selbstbild, soziale Stellung,
 - . Auswahlzwang;
- INFORMATIONELLE FAKTOREN
 - . Portabilität, Kompatibilität, Objektabbildung,
 - . Modularität, Überprüfbarkeit, Transparenz,
 - . Aktualität, Sicherheit,
 - . Syntax, Semantik;
- ORGANISATORISCHE FAKTOREN
 - . Ablauf, Integration,
 - . Strukturierung, Dialogführung, Steuerung,
 - . Verkopplung, Rückkopplung,
 - . Verfügbarkeit, Wirtschaftlichkeit.

Die Berücksichtigung dieser Faktoren geht davon aus, daß die am System beteiligten Komponenten über spezifische Eigenschaften verfügen, die im Verbund auftreten und deren Beachtung unerlässlich ist.

1.3 Zur Charakterisierung des computerunterstützten Unterrichts

Nun einige Thesen zum computerunterstützten Unterricht (CUU), die bereits in den 70er Jahren aufgestellt waren:

- CUU ergänzt die Lehre.
- CUU ermöglicht neue Lehrstrategien, eine Individualisierung des Lernprozesses und eine Lernsteuerung.
- CUU setzt Zusammenarbeit voraus, und zwar zwischen Hochschullehrern, Spezialisten und Studenten.
- CUU setzt Motivation voraus, und zwar sowohl bei Hochschullehrern als auch bei Studenten.
- CUU kann Veränderungen in der Organisation der Lehre zur Folge haben.
- Die Einführung des CUU gleicht einem adaptiven Anpassungsprozess.
- Lehrprogramme im CUU müssen untereinander kompatibel, modular, aktuell und transparent zum Lehrstoff sein.
- CUU-Verfahren müssen einfach in der Entwicklung und im Einsatz, flexibel in der Anpassung und wirtschaftlich in der Anwendung sein.
- Das Kosten-Nutzen-Verhältnis im CUU ist in besonderem Maße von der Einsatzbreite und -häufigkeit (Nutzungsgrad) der Lehrprogramme abhängig.

1.4. Lernstrategie und Programmarten

Bereits die ersten Forschungen auf diesem Gebiet haben zahlreiche Lerntheorien und -strategien sowie Programmarten untersucht. Nicht diskutiert wurde und wird die Notwendigkeit einer Bewertung und daraus einer Diagnose des Kandidaten bzw. seines Wissens; sie gelten als selbstverständlich. Im Normalfall sieht dies folgende Schrittfolge vor:

Eingangstest ---> CUU/CAL/CBT ---> Wissensdiagnose

Dabei werden Eingangstest und Wissensdiagnose einmal oder mehrfach im Laufe einer Sitzung durchgeführt. Die Programme sehen verschiedene Vorkopplungen und Rückkopplungen vor. Schließlich ist die Art und der Inhalt der Aufgabenstellung entscheidend. Aus diesen Fakten resultieren die folgenden Programmarten:

Übung/Training, tutorieller Dialog, Werkzeuggebrauch, Anleitung, Information, Computer-Planspiel, Simulation, Logistik, Problemlösen, Vortest/Nachtest, Wissensbasen, Hypertext.

Die wichtigsten dieser Programmarten seien hier stichpunktartig charakterisiert:

Übung/Drill and Practice

- zur Festigung von aufgenommenen Faktenwissen
- zum Training von Vokabeln, von Rechenverfahren etc.
- Selbst-Test-Hilfe-Programm

Tutorieller Dialog

- verzweigte Lehrprogramme
- schrittweises Lehren
- Aufgabe--->Antwort--->Analyse--->Folgeschritt

Werkzeuggebrauch

- Umgehen mit Tools/Werkzeugen
- Entlastung von Kenntnissen bspw. in der Programmierung

Anleitung

- rechnergeleitetes Selbststudium
- Vermittlung von Lehrstoffen (Computer Managed Instruction) nach persönlichem Kenntnisstand

Information

- lexikalische Bausteine
- Formen der Datenbereitstellung

Simulation

- Nachbildung realer Systeme
- Verfahrensübungen
- Verhaltenssimulation
- Prozeßsimulation

2. Die 80er Jahre - Ernüchterung, Wiederbelebung

2.1 Workstations oder Mainframes?

Den vermeintlichen Vorteilen von Mainframes wie Bereitstellung riesiger Datenmengen zentral unter Wahrung der Datenintegrität, die Datenkonsistenz und Datensicherheit, die Bedienung einer großen Anzahl von Benutzern etc. stehen die Trümpfe von Workstations gegenüber:

- komfortable Bedienungsoberfläche,
- Unabhängigkeit von anderen Benutzern,
- gleichbleibende Antwortzeiten,
- Kosteneinsparungen, effizientes Arbeiten und größere Benutzerakzeptanz.

Diese Vorteile konnten erst jetzt, nach Vordringen der PC's nachgewiesen werden. Hieraus resultieren wesentliche Zielvorgaben des Lernalers:

- Unabhängigkeit, Beteiligung und Eigenverantwortung.

2.2 Computer Based Training

Die Entwicklungen der 80er Jahre sind unter der Bezeichnung Computer Based Training (CBT) bekannt geworden. Diese zeichnen sich durch die in Abbildung 2 dargestellten Charakteristika aus.

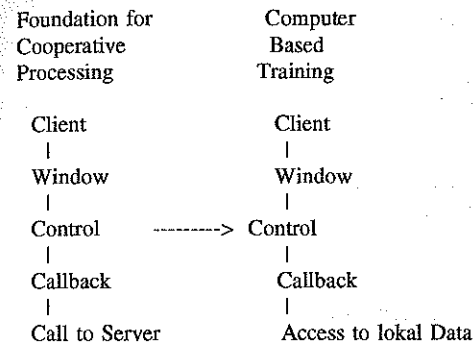


Abb. 2: Prinzip des Computer Based Training

3. Die 90er Jahre - Gegenwart

3.1 Werkzeuge zur Entwicklung von Lern- und Trainingssystemen

Die technologische Gegenwart wird durch folgendes geprägt:

- Auf der Hardware-Seite herrschen Standard-MS-DOS-Rechner vor; zukünftig drängen Unix-Workstations vor, die jedoch prinzipiell keine Änderungen zu den DOS-Rechnern mit sich bringen werden: Sie werden das verbundene Arbeiten besser unterstützen.
- Auf der Software-Seite dominieren dedizierte Werkzeuge. Im Vergleich zum CUU handelt es sich um verbesserte technisch-ergonomisch ausgefeilte, sowie didaktisch-methodisch besser aufbereitete Programme.

Hierbei werden verschiedene Typen von Software-Werkzeugen eingesetzt, die wie folgt untergliedert werden können:

- Programmiersprachen (APL, Pascal, C, Visual Basic)
Sie sind als Werkzeuge nicht spezifisch.
- Autorensprachen (Tutor, TenCORE, TOPIC)
Sie sind als Werkzeuge spezifisch und mitunter sachgebietbezogen.
- Autorensysteme (ETAS, SEF, Mandarin, Authorware)
Sie sind als Werkzeuge nicht flexibel, nicht mächtig genug.

Autorensysteme sind Entwicklungswerkzeuge, mit deren Hilfe qualitativ gute Unterrichtsprogramme mit wirtschaftlich vertretbarem Aufwand erstellt werden sollen. Autorensysteme sollen auch dazu dienen, EDV-Laien die Möglichkeit zur Umsetzung ihrer Lehr-Materialien in Lern-Programme zu geben. Sie sollen die Qualität des Lernprogramms verbessern helfen und die Arbeit des Entwicklers erleichtern. Autorensysteme gehören somit zur Kategorie der Software-Werkzeuge. Entscheidend für ihren Einsatz sind somit ihre Funktionsbereiche und damit ihre Leistungsarten sowie Typenausprägungen.

Für ihre Funktionsbereiche sind folgende Faktoren entscheidend:

- die Präsentation (textlich, bildlich, akustisch),
- die Interaktion (Dialogfunktion, Multiple Choice, Antwortanalysen),
- die Schnittstellen (User Exit).

Aus der Sicht der Typenausprägungen sind hingegen folgende Faktoren wichtig:

- das Sprachsystem,
- die Ab-/Anfragesysteme und
- die integrierten Systeme.

Folgende Beispiele sind charakteristisch:

- Tutor, APL (A Programming Language), SEF/PC (IBM), PILOT, TenCORE, TROPIC,
- CCL (Courseware Construction Language),
- IDE (Instructional Design Environment; LISP, Hypertext),
- IDES (Instructional Design Expert System).

3.2 Komponenten von Lern- und Trainingssystemen

Die Komponenten des Lern- und Trainingssystems sind

- der tutorielle Teil zur Vermittlung logistischer Einzelaufgaben (Entscheidungsprobleme, Wissen --> Informationsvermittlung),
- der Trainings-Teil zum Erlernen von Problemlösungsstrategien
- die multimediale Präsentation zur optischen, akustischen Darstellung mit Video, Ton, Animation, Grafik.

Die Bausteine eines Lernprogramms sind

- die Lern/Lehrziele (Zielstruktur),
- die strukturierten Lern-Inhalte (Elementebene),
- die Präsentation (Text, Bild, Ton, Multimedia),
- die Schnittstellen (Menüs, Bedienungsfläche) und
- die Interaktionen (Lehrstrategie).

Als Interaktionen sind folgende zu nennen:

- die Auswahl (einfach, mehrfach),
- die Zuordnung (allgemein, eindeutig, injektiv, bijektiv),
- die Anordnung (Sonderfall der bijektiven Zuordnung),
- die Freiantwort (Sprachanalyse) und
- die Simulation.

Als Eigenschaften sind hervorzuheben:

- die menüorientierte Benutzeroberfläche,
- das Arbeiten mit Texten und Grafiken,
- die Schnittstellen zu fremden Programmen,
- die Schnittstellen zu anderen Medien und
- die Aufgabe/Frage - Antwort - Analyse.

3.3 Einsatzgebiete von Lern- und Trainingssystemen

Aus der Vielzahl möglicher Einsatzgebiete von Lernprogrammen seien hier genannt:

- die Unterrichtsinformatik, die Berufsbildung, das logistische Wissen (Industrieproduktion, Flugsimulation), die medizinischen Diagnosen (Scanner, Röntgenbild, Befunde), das kaufmännische Verhalten, die Texte/Hypertexte (Bibel) sowie die Softwareschulung.
- Solche Lernprogramme stehen in folgenden Sammlungen bereit:
- IDM Bielefeld, LSW Soest, IPN Kiel.

3.4 Zur Entwicklung von Unterrichtsprogrammen

Die Entwicklung von Unterrichtsprogrammen erfolgt in einem iterativen Entwicklungsprozeß mit folgenden Schritten:

- Analyse des Lehrgebietes (Intelligent Authoring System): KI-Sprachen-Einsatz; Implementierung in C, Pascal;
- Entwurf/Design (Component Display Theory): Entwicklung einer Lernziel-/Inhalts-Matrix;
- Entwicklung (Elaboration Theory) von Wissensstrukturen "top-down";
- Implementation von statistischen und dynamischen Teilen;
- Evaluation, Animation (SuperVGA).

4. Charakterisierung des intelligenten Tutorensystems

Als Beispiel wird nachfolgend die dominierende Form der Unterrichtssoftware charakterisiert. Es ist das intelligente Tutorensystem (ITS). Es unterscheidet sich von konventionellen Systemen des CUU durch mehr Flexibilität, indem es möglichst viel sachliches und pädagogisches Wissen benutzt, das Lehrer für die Unterrichtsgestaltung benötigen, anstatt nur auf der Ebene vorgefertigter Lektionen zu operieren.

4.1 ITS-Generationen

Die erste Generation der ITS versuchte die Lehrer zu simulieren, den Lehrstoff zu präsentieren, Fragen oder Antworten dazu zu stellen und in Abhängigkeit von den Antworten weiterzumachen oder bestimmte Lernsegmente zu wiederholen. Ein solches Programm besteht aus einer Vielzahl von Fenstern: Der Tutand studiert den Inhalt eines Fensters, bedient eine der Tasten, bekommt ein neues Fenster oder ein altes zur Wiederholung. Hierzu eignen sich insbesondere Hypertext- und Hypermediasysteme (Beispiel: HyperCard von Macintosh). Diese Möglichkeiten führten zu einer erneuten "Blütezeit" der computerunterstützten Wissensvermittlung.

Der Nachteil solcher Systeme ist, daß sie keine unvorbereiteten Fragen stellen, der Tutand keine unvorhergesehene Antworten geben und daher auch keine unvorbereiteten Querverweise erhalten kann. Stattdessen muß der Autor/Entwickler alle wichtigen Eventualitäten vorhersehen und im Programm zu berücksichtigen versuchen.

Die zweite Generation der ITS versucht die Programme flexibler zu machen. Anstatt der Vorgabe fertiger Lerneinheiten soll das System in Abhängigkeit von den Antworten des Tutanden einen angepaßten Unterricht ermöglichen. Dies soll mit Hilfe von im Hintergrund agierenden Expertensystemen erfolgen.

4.2 ITS-Modell

Die zweite Generation der ITS erfordert ein Wissensmodell und einen Problemlöser auf der Seite des Unterrichtsstoffes sowie andererseits ein Tutandenmodell über das Wissen des Tutanden. Letzteres muß ein Resultat aus den Aktionen des Tutanden sein. Je exakter dieses Modell ist, desto gezielter kann der Tutand unterstützt werden. Als weitere Komponente kommt die Benutzeroberfläche hinzu. Sie kann sehr aufwendig (graphisch, natürlichsprachlich, multimedial) sein. Hieraus resultiert das in Abbildung 3 gezeigte Schema für das ITS.

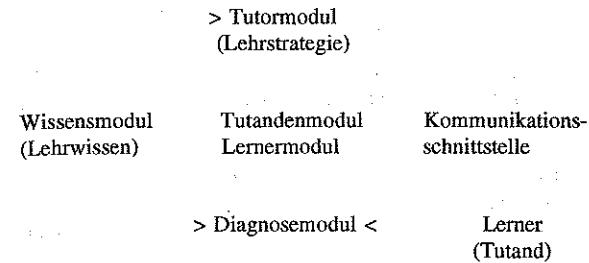


Abb. 3: Schema des intelligenten Tutorensystems

Die Module bestehen aus einer statischen Wissensbasis und einem darauf operierenden Inferenzmechanismus. Aufbau, Inhalte und Techniken basieren auf den Grundprinzipien des CUU:

- Klassische Konzepte wie Übungen, Simulation, generative Verfahren, rechnergeleitetes Selbststudium etc. werden nur technisch verfeinert weitergeführt.
- Neue Varianten wie gemischt-initiativer Dialog, diagnostischer Tutor, Coaching, Mikrowelten sind für gut strukturierte Fachgebiete (Programmiersprachen, Bilanzanalysen) entwickelt.

Das Wissensmodul umschließt die Wissensquelle (-aquisition) und die Bewertungsstandards. Das Wissensmodul muß auch in der Lage sein, Lösungswege und Lösungen des Tutanden nachzuvollziehen und zu vergleichen. Im Prinzip handelt es sich um verschiedene Problemlösungsmethoden und Wissensbasen. Die hier benutzten Modelle gleichen

- semantischen Netzen,
- Frames mit Vererbung,
- Prozeduren zur Pfadsuche,
- vorwärts-rückwärts verketteten Regelsystemen,
- numerischen und qualitativen Simulationssystemen etc.

Das Tutandenmodul modelliert den aktuellen Wissensstand des Tutanden. Um den jeweiligen Wissensstand richtig diagnostizieren zu können, sind nicht die singulären Wissensentitäten, sondern die Wertung des gezeigten Wissens im Kontext von Bedeutung. Aus diesem Grunde muß das Tutandenmodul ablauffähig sein (aktiv), um das erwartete Verhalten mit dem voraussichtlichen Verhalten des Tutanden vergleichen und daraus Aktionen ableiten zu können. In der Praxis haben sich verschiedene Modelle herausgebildet, so bspw.

- Stereotypen (Einordnung des Tutanden nach zusammen auftretenden Eigenschaften in Gruppen);
- Fehlerbibliotheken (Sammlung typischer Fehler des Tutanden);
- Overlay-Modelle (Erweiterung des Wissensmodells um die Antworten des Tutanden; Kennung, Anwendung);
- erweiterte Overlay-Modelle (Eintragung neuer Wissens Elemente auf Grund der Tutanden-Reaktionen);
- konstruktive Tutandenmodelle (Rückschluß auf ein mutmaßliches Wissensmodell aus den vom Tutanden gelösten Fällen).

Die Didaktikkomponente dient dem Dialog des ITS mit dem Tutanden, wobei die eigentliche Umsetzung an der Benutzerschnittstelle erfolgt. Die Dialogkomponente dient einer hohen Lerneffizienz und der Aufrechterhaltung der Motivation des Tutanden. Dabei läuft die Erstpräsentation in Lektionen ab; es folgen Beispiele, Aufgaben mit passenden Schwierigkeitsgraden bis zum selbständigen Problemlösen.

Die Benutzerschnittstelle soll die Diskrepanz zwischen echten Problemsituationen und ihrer Darstellung klein halten und das Vokabular des Tutanden in das interne Vokabular übersetzen. Die Hypermedia-Technologie bietet hierzu viele neue Möglichkeiten. Gebräuchlich sind von Multiple Choice bis zur natürlichsprachigen Dialogschnittstelle.

5. Die Perspektiven

Die absehbaren Entwicklungen deuten auf Verbesserungen der Benutzerfreundlichkeit, des Antwortverhaltens, der Verfügbarkeit, der Zugänglichkeit, der Kommunikationsfähigkeit, der Systemoffenheit etc. hin.

Die Folgen dieser Verbesserungen werden sein, daß zunehmend fachspezifische Software in der selbsterklärenden Form von Lernprogrammen entwickelt wird. Die technischen Möglichkeiten werden viele heute noch vorhandene Grenzen sprengen. Besondere Erfolge werden von den Multi-(Hyper-)media-, Hypertext-, neuronalen Netz- und vernetzten Lernsystemen erwartet.

Probleme können insbesondere dadurch entstehen, daß die Software-Hersteller keine Bemühungen unternehmen, die Grundlagen der Entwicklung von Lehr- bzw. Lernsoftware systematisch zu erforschen und stattdessen halbfertige, optisch gut verkaufbare Systeme auf den Markt werfen.