Wissensbasierte Systeme (WBS)

18.123 - WS 2006/7

Mo	**14:15 - 15:45**	**B-201**
Do	**12:30 - 14:00**	**B-201**

28 Lektionen à 90 min vom 23.10.06 bis 8.2.07

Prof. Bernd Neumann
neumann@informatik.uni-hamburg.de
Sprechzeit Do 16-17h, R-105

Sekretariat Frau Oskarsson, R-107
Tel. 42883-2450
oskarsson@informatik.uni-hamburg.de

Knowledge-based Systems

- Lectures in German, slides in English
- Lectures and slides based on

Computational Intelligence
A Logical Approach

David Poole
Alan Mackworth
Randy Goebel

Oxford University Press, 1998

- PDF versions of slides and other information will be available at

http://kogs-www.informatik.uni-hamburg.de/~neumann/WBS-WS-2006/

Exercises, Projects, Tests

- 1 hour of excercises per week in class
- Participation in a project (to be announced) as "schriftlicher Leistungsnachweis"
- Oral test in February as "mündlicher Leistungsnachweis"

Kleine Einführung in die Künstliche Intelligenz für Schülerinnen und Schüler

- **Seit 1956: "Artificial Intelligence"**
- **Mit dem Rechner Funktionen nachbilden, die zur menschlichen Intelligenz gehören oder intelligente Leistungen ermöglichen**
- **Auch "Intellektik", "Kognitive Informatik", "Rechnerintelligenz"**
- **Etabliertes Forschungsgebiet der Informatik**
  - in Deutschland ca. 75 KI-Professoren
  - Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI) mit ca. 200 Mitarbeitern
  - Zentrum "Intelligente Systeme und Robotik" mit ca. 40 Forschern am Department Informatik der Uni Hamburg

Ziele der KI

KI hat ingenieurwissenschaftliche und kognitionswissenschaftliche Ziele

Ingenieurwissenschaftliche Ziele:

Intelligente Systeme konstruieren

Verbindung zu

- **Ingenieurwissenschaften**
- **Signalverarbeitung**
- **Regelungstechnik**
- **Nanotechnologie**

Kognitionswissenschaftliche Ziele:

Menschliche Intelligenz erklären

Verbindungen zu

- **Neurowissenschaften**
- **Psychologie**
- **Linguistik**
- **Philosophie**

Kerngebiete der KI

Wissensrepräsentation und Wissensverarbeitung

Suchen
Logisches Schließen
Unsicheres Schließen

Sprachverstehen

Verstehen gesprochener Sprache
Textverstehen
Automatische Übersetzung

Bildverstehen

Objekterkennung und Szeneninterpretation
Bildarchivierung

Robotik

Bewegungsplanung
Multiagentensysteme
Kognitive Architekturen

Lernen

Künstliche Neuronale Netze
Konzepte lernen
Selbstorganisation

werden in der LV "Wissensbasierte Systeme" behandelt

Suchen - die primitive Keule der Rechnerintelligenz

Beispiel: Kryptoarithmetische Rätsel

```
  HEMD
+ HOSE
 JACKE
```

A	C	D	E	H	J	K	M	O	S
4	9	0	6	7	1	3	5	2	8

```
 7650
+ 7286
14936
```

Beispiel: Schach

Deep Blue analysiert 200.000.000 Schachstellungen pro Sekunde!

Beispiel: Bildverstehen

Kombination von bedeutungsvollen Formen aus 50.000 Kantenstücken

Schlussfolgern mit Regeln

- **Automatischen Regelauswertung ermöglicht erste Expertensysteme**
  - **Fehlerursachen in technischen Systemen bestimmen**
  - **Medizinische Diagnosehilfen**
  - **Bakteriologische Befunde deuten**
  - **Komplexe Systeme nach Kundenwünschen konfigurieren**
- **Einfache Regelsprachen**
- **Probleme bei Wissensakquisition und Anpassung**

Wie findet ein Expertensystem Fehler in Kraftfahrzeugen?

… z.B. wenn ein Auto nicht anspringt

Regel 1
Wenn(Anlasser arbeitet normal)
dann (Batterie OK)

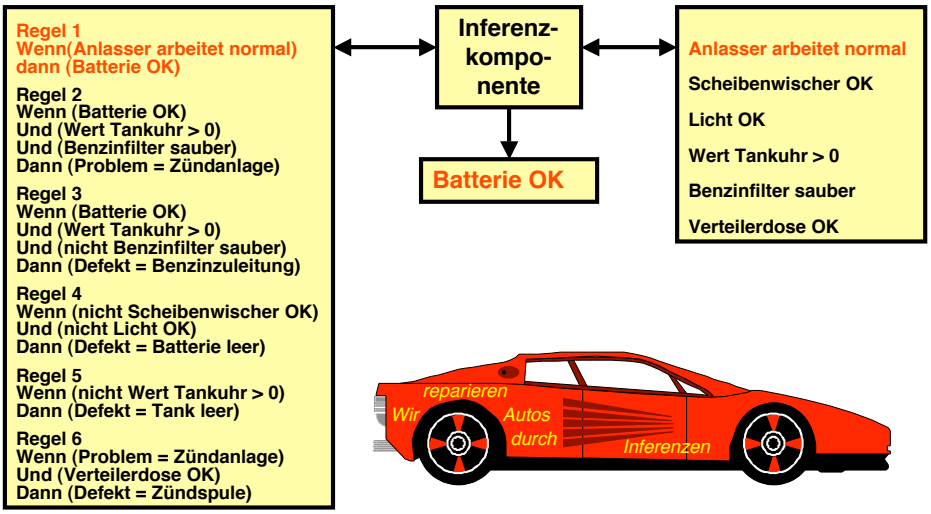
Regel 2
Wenn (Batterie OK)
Und (Wert Tankuhr > 0)
Und (Benzinfilter sauber)
Dann (Problem = Zündanlage)

Regel 3
Wenn (Batterie OK)
Und (Wert Tankuhr > 0)
Und (nicht Benzinfilter sauber)
Dann (Defekt = Benzinzuleitung)

Regel 4
Wenn (nicht Scheibenwischer OK)
Und (nicht Licht OK)
Dann (Defekt = Batterie leer)

Regel 5
Wenn (nicht Wert Tankuhr > 0)
Dann (Defekt = Tank leer)

Regel 6
Wenn (Problem = Zündanlage)
Und (Verteilerdose OK)
Dann (Defekt = Zündspule)



Logik - das Zugpferd der KI

- **Aussagen in einer logisch fundierten Sprache repräsentieren**

  "Haus Angelglück liegt am Plöner See"

  `ferienhaus(HausAngelglück)`
  `see(PlönerSee)`
  `am(HausAngelglück, PlönerSee)`

- **Wissen logisch formulieren**

  "alle Ferienhäuser in der Nähe eines Sees haben Mücken"

  $(\forall X)(\forall Y)\{\ [\texttt{ferienhaus(X)} \wedge \texttt{see(Y)} \wedge \texttt{nahe(X, Y)}] \Rightarrow [\texttt{hatmücken(X)}]\ \}$

  " 'am' bedeutet auch 'nahe' "

  $(\forall X)(\forall Y)\{\ \texttt{am(X, Y)} \Rightarrow \texttt{nahe(X, Y)}\ \}$

- **Schlussfolgerungen aus Aussagen ziehen**

  `hatmücken(HausAngelglück)`

  "Haus Angelglück hat Mücken"

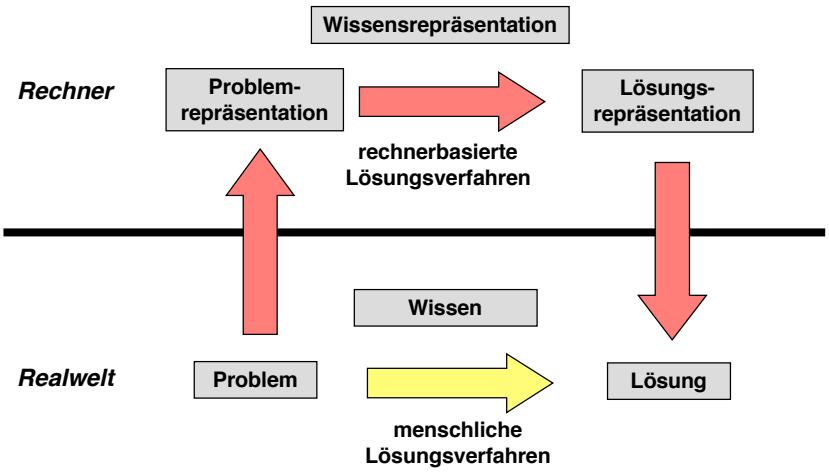
Automatisches Schlussfolgern als Denkprothese

Anwendungen in Problembereichen, deren Komplexität das menschliche Denkvermögen überfordert

<u>Anwendungsbeispiele</u>:

- **Große Datenbanken auf Widerspruchsfreiheit überprüfen**
  Harmonisierung von 150.000 medizinischen Definitionen ergab mehr als 150 Widersprüche
- **Sicherheitskritische Rechnerprogramme auf Korrektheit überprüfen**
- **Ursachen für Fehlfunktionen in komplexen technischen Systemen finden**
- **Komplexe Aggregate unter Berücksichtigung vieler Bedingungen und Regeln konfigurieren**

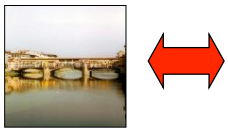
Problemlösen mit einem Wissensbasierten System



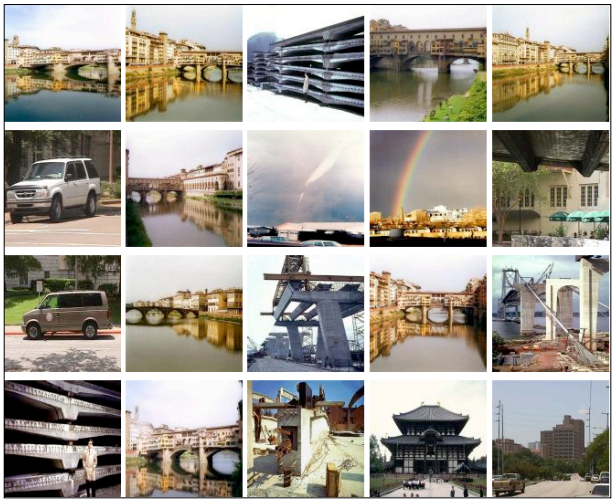
Bildverstehen

- **Gilt als besonders schwieriges Teilgebiet der KI**
- **Menschliches Sehsystem erst teilweise verstanden**
- **Objekterkennung erst in begrenztem Umfang**
- **Dedizierte Systeme für begrenzte Aufgaben**
  - Abruf "ähnlicher" Bilder aus Bildarchiven
  - Fahrerassistenzsysteme im Straßenverkehr
  - Erkennen von kriminellen Handlungen
- **Europäische Forschungsförderung für "Cognitive Vision"**
  - Objektkategorien erkennen
  - Ereignisse erkennen
  - Sehen und Handeln verbinden

Inhaltsbasierter Bildabruf

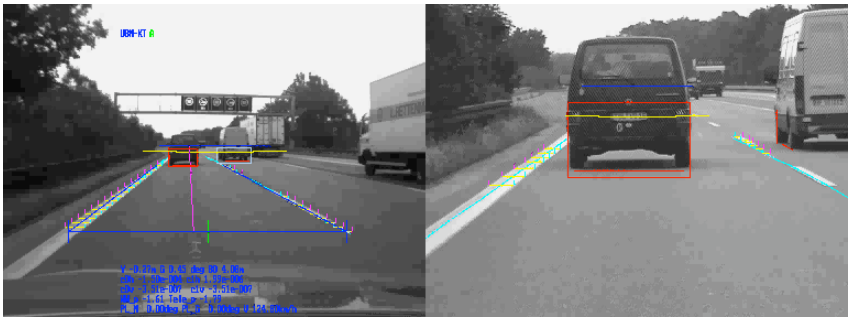


Bildabruf mit CIRES (Content Based Image REtrieval System)



Fahrerassistenzsysteme

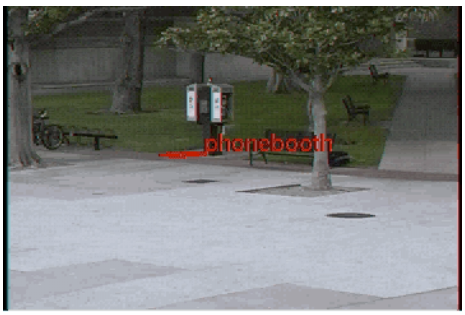
(Dickmanns 1996)



Erkennen krimineller Handlungen

(Hongeng 2003)

Erkennen eines Angriffs



Erkennen eines Diebstahls

Lernen vom Menschen

Wie werten Menschen einzelne Merkmale komplexer Ansichten aus?

Nun wird es ernst!

- **Intelligente Agenten**
- **Symbolverarbeitungshypothese**
- **Wissensverarbeitung in Agenten**
- **Repräsentations- und Schlussfolgerungssysteme**
- **Vorausschau auf Inhalte der Vorlesung**

What is Computational Intelligence?

The study of the design of intelligent agents.

An agent is something that acts in an environment.

An intelligent agent is an agent that acts intelligently:

- its actions are appropriate for its goals and circumstances
- it is flexible to changing environments and goals
- it learns from experience
- it makes appropriate choices given perceptual limitations and finite computation

Central Hypotheses of CI

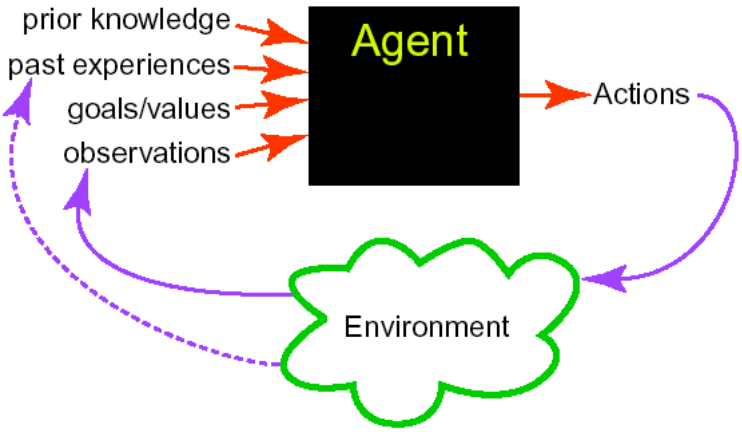
Symbol-system hypothesis:

- Reasoning is symbol manipulation.

Church–Turing thesis:

- Any symbol manipulation can be carried out on a Turing machine.

Agents in the World



Example Agent: Robot

- Actions: movement, grippers, speech, facial expressions, ...
- Observations: vision, sonar, sound, speech recognition, gesture recognition, ...
- Goals: deliver food, rescue people, score goals, explore, ...
- Past experience: effect of steering, slipperiness, how people move, ...
- Prior knowledge: what is an important feature, categories of objects, what a sensor tells us, ...

Example Agent: Teacher

- Actions: present new concept, drill, give test, explain concept, …
- Observations: test results, facial expressions, errors,focus, …
- Goals: particular knowledge, skills, inquisitiveness, social skills, …
- Past experiences: prior test results, effects of teaching strategies, …
- Prior knowledge: subject material, teaching strategies, …

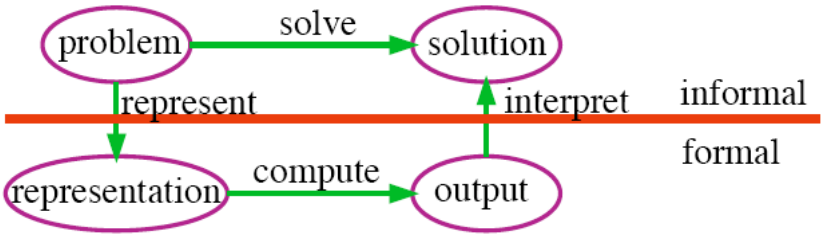
Example Agent: Medical Doctor

- Actions: operate, test, prescribe drugs, explain instructions,…
- Observations: verbal symptoms, test results, visual appearance…
- Goals: remove disease, relieve pain, increase life expectancy, reduce costs,…
- Past experiences: treatment outcomes, effects of drugs, test results given symptoms…
- Prior knowledge: possible diseases, symptoms, possible causal relationships…

Example Agent: User Interface

- actions: present information, ask user, find another information source, filter information, interrupt,…
- observations: users request, information retrieved, user feedback, facial expressions…
- goals: present information, maximize useful information, minimize irrelevant information, privacy,…
- past experiences: effect of presentation modes, reliability of information sources,…
- prior knowledge: information sources, presentation modalities…

Representations



Example representations: machine language, C, Java, Prolog, natural language

What do we Want in a Representation?

We want a representation to be

- rich enough to express the knowledge needed to solve the problem.
- as close to the problem as possible: compact, natural and maintainable.
- amenable to efficient computation; able to express features of the problem we can exploit for computational gain.
- learnable from data and past experiences.
- able to trade off accuracy and computation time.

Representation and Reasoning System

Problem => representation => computation

A representation and reasoning system (RRS) consists of

- Language to communicate with the computer.
- A way to assign meaning to the symbols.
- Procedures to compute answers or solve problems.

Example RRSs:

- Programming languages: Fortran, C++,…
- Natural Language

We want something between these extremes.

Contents Chapter 1

Chapter 1: Computational Intelligence and Knowledge

- Lecture 1 What is computational intelligence?
- Lecture 2 Example application domains and their common features.

Contents Chapters 2 & 3

Chapters 2 & 3: A Representation and Reasoning System

- Lecture 1 Representation and Reasoning Systems. Datalog.
- Lecture 2 Semantics.
- Lecture 3 Variables, queries and answers, limitations.
- Lecture 4 Proofs. Soundness and completeness.
- Lecture 5 SLD resolution.
- Lecture 6 Proofs with variables. Function Symbols.

Contents Chapter 4

Chapter 4: Searching

- Lecture 1 Searching. Graphs. Generic search engine.
- Lecture 2 Blind search strategies.
- Lecture 3 Heuristic search, including *A*_.
- Lecture 4 Pruning the search space, direction of search, iterative deepening, dynamic programming.
- Lecture 5 Constraint satisfaction problems, consistency algorithms.
- Lecture 6 Hill climbing, randomized algorithms.

Contents Chapter 5

Chapter 5: Representing Knowledge

- Lecture 1 Knowledge representation issues. Defining a solution. Choosing a representation. Mapping from problem to a representation.
- Lecture 2 Choosing objects and relations. Semantic networks, frames, primitive and derived relations.
- Lecture 3 Knowledge sharing, ontologies.

Contents Chapter 6

Chapter 6: Knowledge Engineering

- Lecture 1 Knowledge-based systems, roles of people involved, implementing KBSs: base and metalanguages.
- Lecture 2 Vanilla meta-interpreter, depth-bounded and delaying meta-interpreters.
- Lecture 3 Users. Ask-the-user.
- Lecture 4 Explanation and knowledge-based debugging tools.

Contents Chapter 7

Chapter 7: Beyond Definite Knowledge

- Lecture 1 Equality, inequality and the unique names assumption
- Lecture 2 Complete knowledge assumption and negation as failure.
- Lecture 3 Integrity Constraints, consistency-based diagnosis.

Contents Chapter 8

Chapter 8: Actions and Planning

- Lecture 1 Actions, planning and the robot planning domain
- Lecture 2 The STRIPS representation
- Lecture 3 The situation calculus.
- Lecture 4 Planning, forward and resolution planning.
- Lecture 5 The STRIPS planner.
- Lecture 6 Regression planner.

Contents Chapter 9

Chapter 9: Assumption-based Reasoning

- Lecture 1 Assumption-based reasoning framework.
- Lecture 2 Default reasoning, the multiple-extension problem, skeptical reasoning.
- Lecture 3 Abduction, abductibe diagnosis
- Lecture 4 Combining Evidential and Causal Reasoning
- Lecture 5 Algorithms

Contents Chapter 10

Chapter 10: Using Uncertain Knowledge

- Lecture 1 Uncertainty and Probability
- Lecture 2 Conditional Independence and Belief Networks
- Lecture 3 Understanding Independence
- Lecture 4 Probabilistic Inference
- Lecture 5 Markov Chains and Hidden Markov Models
- Lecture 6 Making Decisions Under Uncertainty

Contents Chapter 11

Chapter 11: Learning

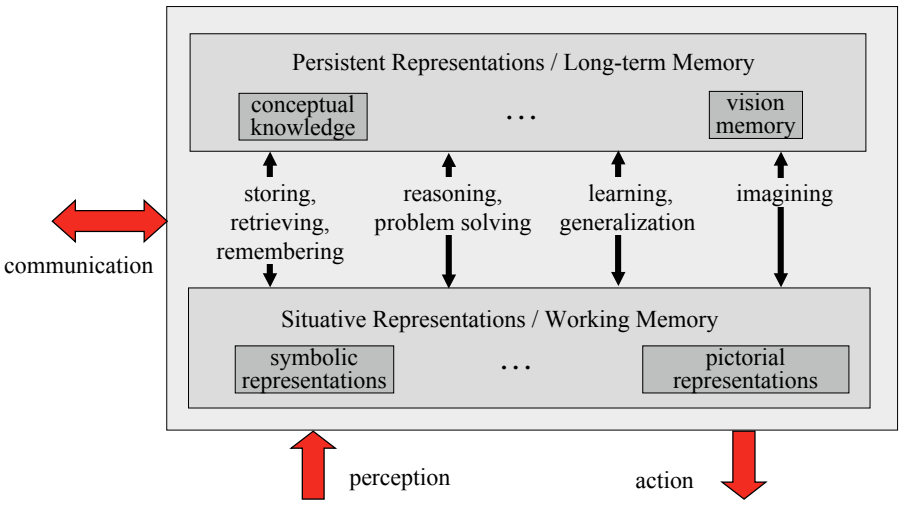
- Lecture 1 Learning Issues.
- Lecture 2 Decision-tree learning.
- Lecture 3 Neural network learning.
- Lecture 4 Case-Based reasoning.
- Lecture 5 Learning under uncertainty.

Contents Chapter 12

Chapter 12: Building Situated Robots

- Lecture 1 Situated robots, robotic systems, robot controllers.
- Lecture 2 Robot architectures and hierarchical decompositions.

Is This All of Artificial Intelligence?



Important AI Areas not covered in WBS

- Natural Language and Speech Understanding
  semantics, translation, abstracting, web retrieval, ...
- Computer Vision
  image analysis, sensor fusion, pattern recognition, object recognition, scene interpretation, image retrieval, document analysis, ...
- Human Cognition
  neural architecture, human perception, spatial and temporal modelling, user models, ...

CI in "Grand Challenge" EU Projects

(Draft of the Information Society Technologies Advisory Group, July 2004)

1. The 100% Safe Car
2. The Multilingual Companion
3. The Service Robot Companion
4. The Self-Monitoring and Self-Repairing Computer
5. The Internet Police Agent
6. The Disease and Treatment Simulator
7. The Augmented Personal Memory
8. The Pervasive Communication Jacket
9. The Personal Everywhere Visualiser
10. The Ultra-light Aerial Transport Agent
11. The Intelligent Retail Store