

# Dr. Michael Thess

---

prudsys AG

**Vortrag auf den 7. DATA-MINING-CUP® Anwendertagen**  
(Chemnitz vom 27.-29. Juni 2006)



Copyright-Hinweis:  
Das Urheberrecht des folgenden Vortrags  
liegt beim Autor. Verbreitung, Vervielfäl-  
tigung und Kopie, auch auszugsweise,  
ist nur mit schriftlicher Genehmigung  
des Autors erlaubt.



## Das Geheimnis adaptiver Empfehlungen im Handel DMC Anwendertage 2006

Michael Thess, prudsys AG

### Inhalt



Echtzeitapplikationen im Data Mining

- I. Adaptives Data Mining
- II. Empfehlungsmaschinen
- III. Reinforcement Learning
- IV. Offline-Modus
- V. Online-Modus
- VI. Beispiel Buch24
- VII. Zusammenfassung

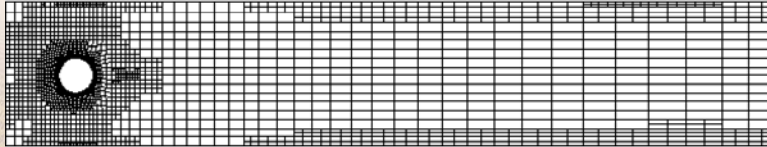


- **Adaptivität: Anpassung eines Systems über mehrere Schritte hinweg**
- **Vorteile:**
  - Weniger Voraussetzungen an das System erforderlich
  - System kann sich an wechselnde Bedingungen anpassen (instationäre Prozesse)
- **Nachteile:**
  - Erfordert erweiterte Infrastruktur, um Rückkopplung zu realisieren
  - Komplexere Theorie der Modellierung
- **Adaptives Data Mining:**
  - Lerner über Interaktion
  - Führt zu Shift im Verständnis von Data Mining:
    - Früher: „Je mehr historische Daten, desto besser“ (im Sinne der Statistik immer noch korrekt)
    - Jetzt: „Wichtiger als Auswertung rein historischer Daten ist das Lernen über Interaktion“
    - Optimal: beides kombinieren (Initialisierung der Modelle durch Lernen aus historischen Daten, danach Verbesserung der Modelle durch Interaktion, also Adaptivität)
  - Vorteile wie oben, insbesondere weniger statistische Annahmen
  - Nachteile wie oben, insbesondere ständige Kommunikation des Data-Mining-Systems mit Kunden erfordert neue, Service-orientierte Infrastruktur
  - Günstig: Entwicklung der IT-Infrastruktur von Unternehmen geht in diese Richtung (z.B. SOA)!

- **Trend zur Adaptivität nicht nur im Data Mining**
- **Betrachten zum Beispiel Lösung von Differentialgleichungen**
  - geschieht zumeist über numerische Verfahren
  - wichtigstes Verfahren: Finite Elemente Methode (FEM)
  - Idee: Lösungsfunktion  $f$  wird über Ansatzfunktionen eines Gitters aufgebaut, je dichter das Gitter, desto besser die Approximation (Konvergenz)
  - Konvergenz und Konvergenzgeschwindigkeit von Regularität des Problems abhängig, insbesondere von Form des Randes (Lipschitz-Bedingung, uniforme Kegelbedingung, etc.) und spezifischen Randbedingungen sowie Lastverteilung
- **Klassische FEM-Ansätze:**
  - „übliche“ Regularitäts-Bedingungen werden vorausgesetzt, weil meist erfüllt
  - Iterationsverfahren zur FEM-Lösung konvergieren schnell
- **Praxis:**
  - „böartige“ Singularitäten, komplexe Randbedingungen
  - Verformtes Werkstück wirklich mit Lipschitz-stetigen Rändern? Presslast wirklich Dirac-sche Delta-Distribution oder doch eher aus Sobolevraum zweiter Ordnung? Wissen wir das? Woher?
  - Ergebnis: für komplexe Probleme oft schlechte Konvergenz mit ungenauen Resultaten

Motivation: FEM

- Daher Trend zu adaptiven FEM-Ansätzen
- Prinzipielle Vorgehensweise:
  1. Erzeugung eines initialen Gitters
  2. Lösung des FEM-Systems auf aktuellem Gitter
  3. Berechnung des Fehlers über A-Posteriori Fehlerschätzer
  4. Wenn Fehler kleiner vorgegebener Schranke → Stop
  5. Verfeinerung des Gitters unter Nutzung der realen Geometrie an Stellen mit hohen Fehleranteilen
  6. Gehe zu 2.
- Ergebnis: zügige Konvergenz, Gitter erfordern relativ wenig Knoten, Knoten gruppieren sich zumeist um Singularitäten



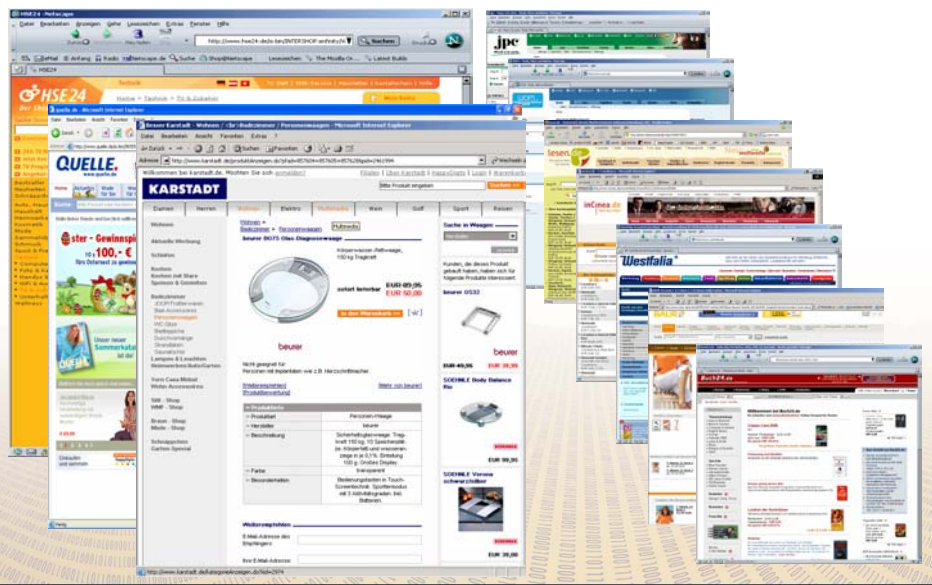
Überblick

- Empfehlungsmaschinen (Recommendation Engines, RE) unterbreiten automatisiert Empfehlungen am POS
- Vorreiter: Amazon.com; wer ein Buch anschaut erhält passende Bücher als Empfehlungen
- Basieren zumeist auf automatischen Analysen des Kundenverhaltens bzw. der Produkte
- Zahlreiche Varianten und Technologien zur Realisierung von REs, Thema intensiver Forschung
- Ziel: Umsatzerhöhung, aber auch zahlreiche erweiterte Funktionen wie Preis- und Sortimentsoptimierung
- Optimal für adaptives Data Mining, da sie mit Kunden interagieren
- prudsys Vorreiter und Marktführer von REs in Deutschland, zentrale Technologie der prudsys RE - Reinforcement Learning (siehe hinten):
  - Ausgangspunkt: REs für Webshops
  - derzeit weitere Anwendungen: Content-Management-Systeme, Self-Service-Systeme, Call-Center
  - Besonderer Schwerpunkt: REs im stationären Handel; erfordern völlig neuartige Interaktionsplattformen, diese werden von der PIRONET Group entwickelt

## Anwendungsgebiete der prudsys RE



Referenzen E-Commerce: Webshops



prudsys AG, DMC 2006

Page 7

## Anwendungsgebiete der prudsys RE



Neue Applikationen im stationären Handel – in Zusammenarbeit mit der PIRONET Group



**Applikation:** Personal Shopping Assistant (PSA)  
**Anwendung:** beim Scannen von Produkten werden Empfehlungen angezeigt  
**Partner:** PIRONET NDH  
**System:** Hardware: Wincor Nixdorf, Software: PIRONET NDH  
**PSA-Referenzen:** Metro Future Store, Metro Cash&Carry, Dohle HIT Vision Store



**Applikation:** Infoterminals XPLACE  
**Anwendung:** beim Scannen / Info von Produkten werden Empfehlungen angezeigt  
**Partner:** Advertimes / Advertools  
**System:** Hardware: Adverttools, Software: Adverttools  
**XPLACE-Referenzen:** Saturn, Media Markt, Thalia



**Applikation:** Checkout-Couponing  
**Anwendung:** individueller Coupon-Druck an der Kasse  
**Partner:** Acardo  
**System:** Hardware: verschiedene Kassenanbieter, Software: Acardo  
**C-Couponing-Referenzen:** Edeka

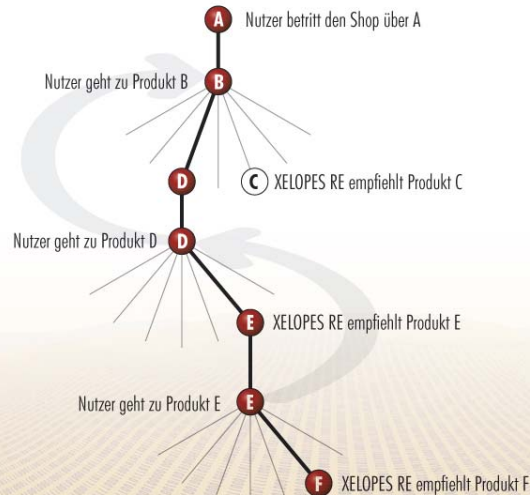
Weitere Applikationen in Vorbereitung....

prudsys AG, DMC 2006

Page 8



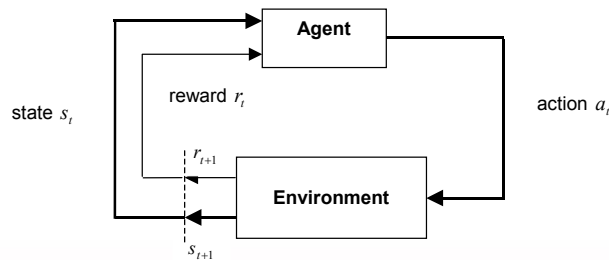
Motivation: Arbeitsweise selbstlernende Empfehlungsmaschine



Modellierung

- Zentrales Framework der prudsys RE
- Ausführlich vorgestellt auf DMC 2005, „Echtzeitanwendungen im Data Mining“
- MDP (Markov Decision Process)-basierter Ansatz
- Mathematische Formulierung: Lösung der Bellman-Gleichung
- Vorgehen:
  - Data Mining auf historischen Daten zur Schätzung der Übergangswahrscheinlichkeiten
  - Lösung der Bellman-Gleichung mittels Dynamischer Programmierung
  - danach Nutzung von online-Verfahren (Monte Carlo, Temporal-Difference Learning) zum Nachlernen und für Echtzeitempfehlungen
  - Nutzung der Aktions-Werte-Funktion zur Simulation
  - Optimierung von Dimensionsgröße unter Nutzung der Aktions-Werte-Funktion
- Zusammenfassend:
  - nahtlose Kombination grundlegender Reinforcement-Learning- und Data-Mining-Verfahren zum kombinierten offline- und online-Lernen und Simulation

- Inhalt: Selbstlernen mittels Wechselspiel von Lernen und Handeln
- Zentrale Ansätze:
  - Lernen geschieht über mehrere Etappen, mit Belohnung (Reward) je Etappe
  - dabei soll Zielgröße Reward maximiert werden
  - Wechselspiel „exploit“ <---> „explore“
- Anwendungen:
  - Robotersteuerungen
  - Spiele (Schach, Backgammon, Blackjack,...)
  - Agentensteuerungen, Finanz-Simulationen
- Grundlegende Begriffe im RL:
  - Zustand:  $s$  (state)
  - Aktion:  $a$  (action)
  - Belohnung:  $r$  (reward), soll über alle Zustände maximiert werden
  - Übergangswahrscheinlichkeiten:  $p(s_1, s_2, a)$
  - Übergangsbewohnungen:  $r(s_1, s_2, a)$
  - Zustands-Wert-Funktion:  $V(s)$  (state-value function)
  - Aktions-Wert-Funktion:  $Q(s, a)$  (action-value function)
  - gesucht: Policy  $\pi$ , stochastisch:  $\pi(s, a)$  oder deterministisch  $a = \pi(s)$



- Beschreibung als Wechselwirkung von Agent / Environment
- Voraussetzung:
  - Markov-Property: optimale Policy hängt nur von aktuellem Zustand  $s$  ab!
  - Klassisches Beispiel: Schach, bester Zug nur von aktueller Position abhängig
  - Anwendbarkeit der Markov-Property muß für jede Applikation geprüft werden
  - RL-Aufgabe, welche Markov-Property erfüllt, heißt Markov Decision Process (MDP)
  - betrachten hier nur MDPs

- Idee: RE lernt jedesmal neu auf Basis aller historischen Transaktionsdaten
  - Typische Transaktionsdaten:
    - Aufträge (im Webshop, Kassensystemen, Call Center)
    - Surfdaten (Clickstreamdaten aus Logdateien, spezielle Tabellen)
    - Suchdaten, etc.
  - Vorgehensweise:
    - Definition einer Kennziffer für den Reward  $r$
    - Data Mining (Warenkorb- und Sequenzanalyse, Regression) zur Ermittlung der Übergangswahrscheinlichkeiten  $p$
    - Lösung der Bellman-Gleichung
    - Nutzung der Policy  $\pi_i$  für Empfehlungen
  - Vor- und Nachteile:
    - Vorteile: leichte Integrierbarkeit, historische Daten können genutzt werden
    - Nachteile: geringere Qualität, langsame Anpassung an Nutzer- und Produktänderungen
- Offline-Modus sollte zu Beginn der Inbetriebnahme der RE genutzt werden

- Idee: RE lernt über Interaktion mit Nutzern
  - Vorgehensweise:
    - Definition einer Kennziffer für den Reward  $r$
    - Einlesen der bisherigen Policy  $\pi_i$  einschließlich Aktions-Werte-Funktion  $Q$
    - Starten der RE im Server-Modus (RE enthält integrierten Webserver)
    - Empfehlungen und deren Update (von  $Q$  und  $\pi_i$ ) über Interaktion mit RE, Nutzung von Online-RL-Algorithmen zum Update
    - Runterfahren der RE, aktuelle Policy  $\pi_i$  wird gespeichert
  - Vor- und Nachteile:
    - Vorteile: hohe Qualität, zahlreiche neue Funktionen realisierbar
    - Nachteile: komplexere Anbindung
- Online-Modus sollte auf den Offline-Modus folgen, mit Offline-Policy als Start-Policy
- Um Komplexität des Online-Modus zu umgehen, Kompromiß – Batch-Online Modus:
  - Technisch ähnlich einfach wie Offline-Modus
  - Speichert Feedback der Empfehlungen in spezieller Tabelle
  - Führt regelmäßiges Update der Policy unter Nutzung von Online-artigen Algorithmen



- URL: [www.buch24.de](http://www.buch24.de)
- Erfolgreicher Webshop mit über 1.5 Mio. Artikeln im Angebot
- Komplexität des Angebotes stellt RE vor zahlreiche Schwierigkeiten
- Günstig: eigenes, vollständig dynamisches Shopsystem



- Systematische Anbindung der prudsys RE:
  1. Offline-Modus
  2. Batch-Online Modus
  3. Online-Modus

- Ausgangspunkt nach Installation der prudsys RE
- Transaktionsdaten: Aufträge, Clickstreams
- Stammdaten: sehr umfangreich
- Definition geeigneter Kategorien (über Autoren, Buchklassifikationen, etc.), da RE intensiv mit Kategorien arbeitet; garantiert gute Empfehlungen für nahezu sämtliche Produkte
- Definition eines geeigneten Reward: komplexe Kennziffer, welche Preise, Erscheinungsdaten, Lagerbestand, etc. enthält
- prudsys RE generierte regelmäßig Policy in Form von Regeln unter Nutzung der Stamm- und Transaktionsdaten sowie des Reward
- Regeln in Datenbank gespeichert und als Empfehlungen angezeigt
- Installation eines Kontrollgruppen-basierten Mechanismus zur Anzeige unterschiedlicher Regelarten sowie für Erfolgsmessung
- Auswertung des Erfolgs unter Nutzung des OLAP-Tools MERKUR MINER PLUS

- Ziel: schrittweiser Übergang zum Batch-Online Modus
- Nutzung einer speziellen Feedback-Tabelle, welche bereits von Buch24 angelegt war, Erweiterung der Tabelle für Batch-Online Modus der prudsys RE
- Feedback-Tabelle speichert wichtige Informationen über produktbezogene Clickstreams, Warenkörbe, Aufträge sowie angezeigte Empfehlungen und deren Feedback
- Konfiguration der prudsys RE für Batch-Online Modus
- Initiale Regeln: Bewährte Regeln aus Offline-Modus
- prudsys RE lädt alle 2-3 Tage bisherige Regeln aus DB, berechnet auf Basis der Feedback-Tabelle (sowie Stammdaten und Reward) deren Update, schreibt Regeln in DB zurück, danach Deployment im Shop
- Implementierung eines speziellen Empfehlungsmechanismus, welcher basierend auf Kennzahlen der Regeln diese zur Anzeige bringt und zugleich explorative Empfehlungen gleitend einstreut

- Vollständiger Online-Modus
  - Nutzt dynamische Fähigkeiten des Shopsystems
  - Konfiguration der prudsys RE für Batch-Online Modus
  - Initiale Regeln: Bewährte Regeln aus Batch-Online-Modus
  - prudsys RE wird im Online-Modus gestartet: Liest initiale Regeln aus Datenbank, startet internen Webserver, danach Kommunikation mit Shop über spezielle SOAP-Schnittstellen, dabei Update der Regeln (einschließlich Nutzung der Stammdaten und des Reward)
  - prudsys RE wird runtergefahren und speichert Regeln wieder in DB ab
  - derzeit im Testbetrieb; in Kürze online
- Empfehlungen in höchster Qualität
- Weitere umfangreiche Funktionen in Vorbereitung (insbesondere im Bereich Personalisierung)

- **Adaptives Data Mining bedeutet Shift im Verständnis von Data Mining:**
  - Früher: Lernen statisch auf Basis historischer Daten
  - Künftig: Lernen dynamisch durch Interaktion
  - Beides kann sinnvoll kombiniert werden
  - Ergebnis: Adaptive Data-Mining-Systeme
- **Empfehlungsmaschinen:**
  - Optimale Plattform und Voraussetzung für adaptives Data Mining
  - Erfordert service-orientierte IT-Infrastruktur, wird derzeit in vielen Unternehmen entwickelt
  - Reinforcement Learning perfekte Theorie für adaptive, instationäre Prozesse
- **Vorgehen:**
  - Schrittweises und systematisches Vorgehen ist Voraussetzung
  - Gekennzeichnet durch Übergang von Offline- zu Online-Modus
- **Vorteile:**
  - Höchste Qualität der Ergebnisse
  - Schnelle Anpassung an veränderte Rahmenbedingungen
  - Völlig neuartige Funktionalitäten für Personalisierung im Online-Modus