Freie Universität Berlin

STI · BERLIN

State-of-the-art on Recommender Systems

Überblick über die klassischen Verfahren und semantische Erweiterungen

Radoslaw Oldakowski
FU Berlin

Marko Harasic
T-Systems MMS GmbH

Corporate Semantic Web Workshop, Xinnovations 2010, Berlin, 19.09.2011

INNOPROFILE UNTERNEHMEN REGION
Die BMBF-Innovationsinitiative Neue Länder

T-Systems

CSW Corporate Semantic Web

Agenda

- Teil I: Klassische Recommender-Ansätze
  - Collaborative Filtering
  - Content-based Recommender
  - Knowledge-based Recommender
  - Hybride Recommender

- Teil II: Semantische Erweiterungen
  - Definition Semantische Recommender
  - Konzept-basierte Recommender
  - Taxonomische Recommender
  - Erweiterungspotential

Einführung

- Recommender-Systeme als adaptive Systeme
  - Automatisieren den (online) Entscheidungsprozess
    - Nutzerpräferenzen
    - Präferenzen ähnlicher Nutzer
    - Expertenwissen
  - Passen ihr Verhalten an den individuellen Nutzer an

Collaborative Filtering (CF): Grundidee

- Sehr gut erforscht seit Mitte der 90-er Jahre
- Weite Verbreitung in kommerziellen Anwendungen (Bsp. Amazon)
- Generierung von personalisierten Empfehlungen auf Basis von historischen Daten einer Nutzer-Community
- Input: Bewertungsmatrix
  - Explizite vs. implizite Bewertungen
- Output:
  - numerische Vorhersage
  - Liste von N Empfehlungen

	E1	E2	E3	E4	E5
N_x	5	3	4	4	???
N_1	3	1	2	3	3
N_2	4	3	4	3	5
N_3	3	3	1	5	4
N_4	1	5	5	2	1

CF: User-based Nearest Neighbor Recommendation

- Suche nach **ähnlichen Nutzern** (mit gleichen Verhaltensmustern)
  - Korrelationskoeffizient (nach Pearson)

$$sim(a,b) = \frac{\sum_{e \in E}(r_{a,e} - \bar{r}_a)(r_{b,e} - \bar{r}_b)}{\sqrt{\sum_{e \in E}(r_{a,e} - \bar{r}_a)^2}\sqrt{\sum_{e \in E}(r_{b,e} - \bar{r}_b)^2}}$$

  - z.B. $sim(N_x, N_1) = 0{,}85$; $sim(N_x, N_2) = 0{,}7$

	E1	E2	E3	E4	E5
N_x	5	3	4	4	???
N_1	3	1	2	3	3
N_2	4	3	4	3	5
N_3	3	3	1	5	4
N_4	1	5	5	2	1

- Verwendung der Ähnlichkeiten zwischen Nutzern zur Generierung der **Vorhersage für den aktiven Nutzer**

$$pred(a,e) = \bar{r}_a + \frac{\sum_{b \in N} sim(a,b)(r_{b,e} - \bar{r}_b)}{\sum_{b \in N} sim(a,b)}$$

  - z.B. $pred(N_x, E_5) = 4{,}87$

- Memory-based → ungeeignet für große Bewertungsmatrizen

CF: Item-based Nearest Neighbour Recommendation

- basieret auf der **Ähnlichkeit zwischen Elementen**
  - *adjusted cosine similarity*
  - z.B. $sim(E_5,E_1) = 0{,}80$; $sim(E_5,E_4) = 0{,}41$

	E1	E2	E3	E4	E5
N_x	5	3	4	4	???
N_1	3	1	2	3	3
N_2	4	3	4	3	5
N_3	3	3	1	5	4
N_4	1	5	5	2	1

- Berechnung der Vorhersage als gewichtete Summe der Nutzerbewertungen von ähnlichen Elemente

$$pred(u,e) = \frac{\sum_{i \in ratedItems(u)} sim(i,e) * r_{u,i}}{\sum_{i \in ratedItems(u)} sim(i,e)}$$

  - $pred(N_x,E_5) = 4{,}66$

- Model-based
  - geeignet für offline-preprocessing
  - Berechnung der Vorhersagen in real-time

CF: Vor- und Nachteile

+ Gute Qualität und Performance
+ keine zusätzlichen Informationen über Nutzer und Elemente notwendig
+ subjektive Bewertungsfaktoren werden berücksichtigt
+ Serendipität

- Große Nutzer-Community erforderlich
- Data-Sparsity-Problem
- Cold-Start-Problem
  - Neue Nutzer
  - Neue Elemente

	E1	E2	E3	E4	E5
N_x			???		
N_1	3	1		3	3
N_2		3		3	
N_3	3				4
N_4	1	5		2	1

Content-based Recommender (CBR): Grundidee

- Generierung von Empfehlungen durch Matching von Nutzerpräferenzen und Elementeigenschaften
  - Anwendungsbereiche
    - Nachrichten, Webseiten, Produkte, Restaurants, Filme, Fernsehsendungen
- Voraussetzung
  - Beschreibungen der Eigenschaften von Elementen (Content)
    - Strukturierte (Meta-)Daten

Titel	Zustand	Sprache	Erscheinungsjahr	Produktart	Genre	Subgenre
Physik Formeln und Gesetze	Neu	Deutsch	2009	Lernhilfe	Wissen & Technik	Physik

    - Extraktion aus Textbeschreibungen
  - Informationen über Nutzer (Nutzerprofil)
    - History
    - Präferenzen: implizit vs. explizit
- Vergleichsmechanismen
  - Ähnlichkeitsbasiert, probabilistische Ansätze, Machine Learning Verfahren

Beispiel: CBR bei unstrukturierten Daten wie Texte, Webseiten

- Fehlende Attributbezeichnung
  - jedes Wort als Elementeigenschaft
- Elementrepräsentation
  - *Vector space model*
  - Preprocessing:
    - Entfernung von Stoppwörter, Stemming, size cuttofs, Phrasen
    - Gewichtung: TF-IDF
- Profilrepräsentation
  - *Vector space model*
    - z.B. Durchschnittsvektor aller Elementvektoren aus der Nutzer-History
- Berechnung der Empfehlungen
  - Kosinus-Ähnlichkeit

$$sim(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| * |\vec{b}|}$$

CBR: Vor- und Nachteile

+ Große Nutzer-Community ist nicht notwendig
+ Kein New-Item-Problem
+ Geeignet für Elemente von kurzer Lebensdauer

- Elementbeschreibungen müssen vorhanden sein
  - Extraktion kann problematisch sein
- Cold-Start-Problem
  - Neue Nutzer
- Überspezialisierung
- Stabilitätsproblem

Knowledge-based Recommender (KBR)

- Generierung von Empfehlungen auf Basis von
  - Nutzeranforderungen
  - Elementeigenschaften
  - zusätzlichem domänespezifischem Wissen

  } **Wissensbasis**

- Interaktivität (*conversational systems*)
  - Führen den individuellen Nutzer zu relevanten Elementen in einer großen Menge an Optionen (Bsp. *critiquing*)

- Besonders geeignet für
  - Komplexe Produkte mit vielen Eigenschaften (z.B. Unterhaltungselektronik)
  - Selten gekaufte Produkte (z.B. Autos, Finanzdienstleistungen)

- Arten von KBR Systemen
  - **constraint-based** – vordefinierte Empfehlungsregeln
    - *constraint satisfaction problem* → constraint solver oder konjunktive Anfragen
  - **case-based** – verschieden Ähnlichkeitsmaße
    - z.B. Distanzfunktionen

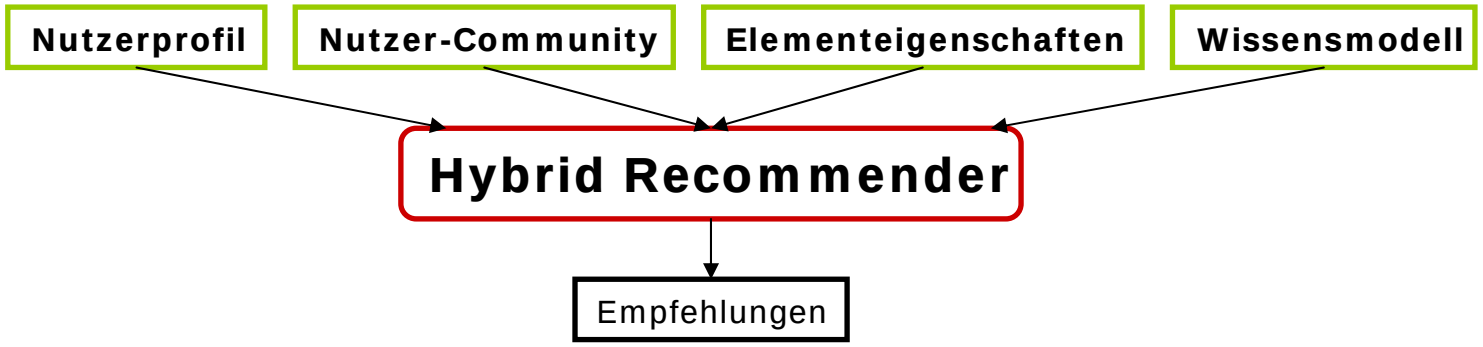
KBR: Vor- und Nachteile

+ Domänewissen fließt in die Empfehlungen ein
+ Kein Cold-Start-Problem
+ Änderungen der Präferenzen werden sofort berücksichtigt

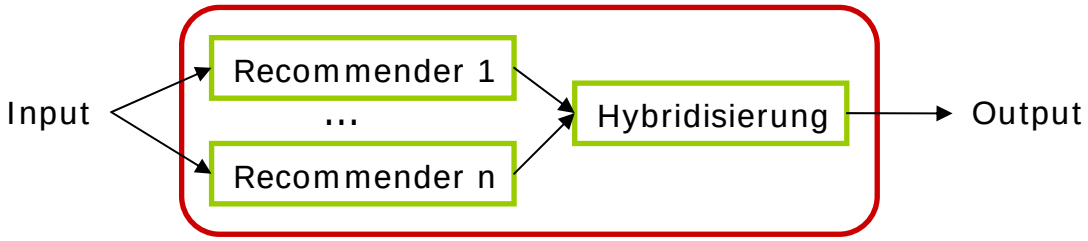
- Erstellung und Pflege der Wissensbasis
  - Empfehlungsqualität hängt stark von der Wissensbasis ab
- Wissensbasis ist hartcodiert im System
- Stonewalling
  - Bei zu restriktiven Anforderungen → keine Ergebnisse
  - Die dafür verantwortlichen Anforderungen sind manchmal schwer zu ermitteln

Hybride Recommender-Systeme (HRS)

- Kombination verschiedener Datenquellen und Recommender-Ansätze

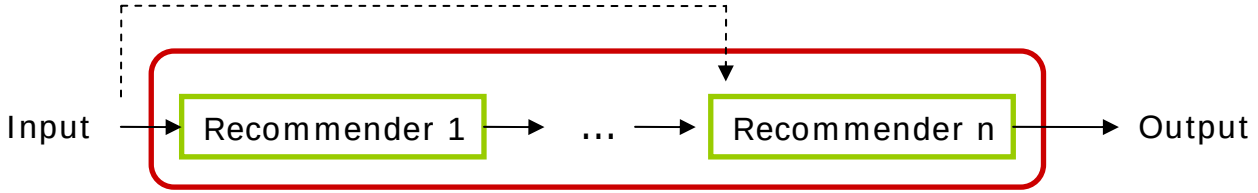


- Parallelized
  - switching
  - mixed
  - weighted

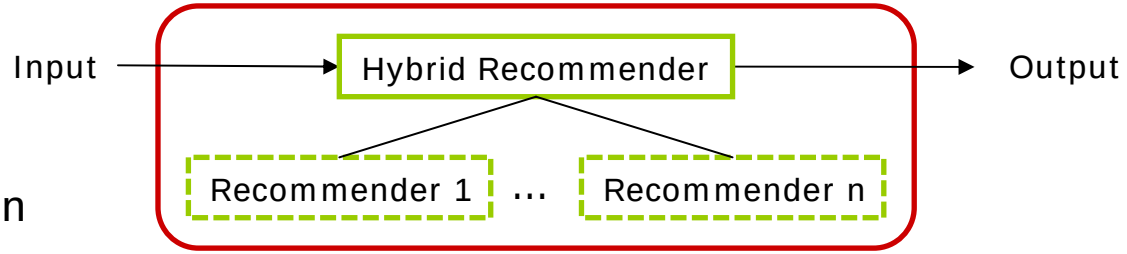


Hybride Recommender-Systeme

- Pipelined
  - meta-level
  - cascade



- Monolithic
  - feature combination
  - feature augmentation



Semantische Recommender

- Teil I: Klassische Recommender-Ansätze
  - Einführung
  - Collaborative Filtering
  - Content-based Recommender
  - Knowledge-based Recommender
  - Hybride Recommender

- Teil II: Semantische Erweiterungen
  - Definition Semantische Recommender
  - Konzept-basierte Recommender
  - Taxonomische Recommender
  - Erweiterungspotential

Was sind Semantische Recommender

- Hybrid eines „klassischen“ Recommenders mit einem Knowledge-Based Recommender
- Alle Recommender Systeme mit einer Taxonomie oder Ontologie als Wissensbasis [Peis et al., 2008]
- Erweiterung eines CBR durch semantische Vergleiche
- Erweiterung von CF durch Einbeziehung von semantischen Benutzerähnlichkeiten anhand ihrer Vorlieben
- Algorithmen arbeiten anhand einer modularen Wissensbasis

Wissensbasis

- Ontologien in RDF / OWL stellen Wissensbasis bereit
- Ontologien dabei oftmals bereits vorhanden
  - Von Domänenexperten erzeugt / gewartet
  - Aktualisierungen sofort integrierbar

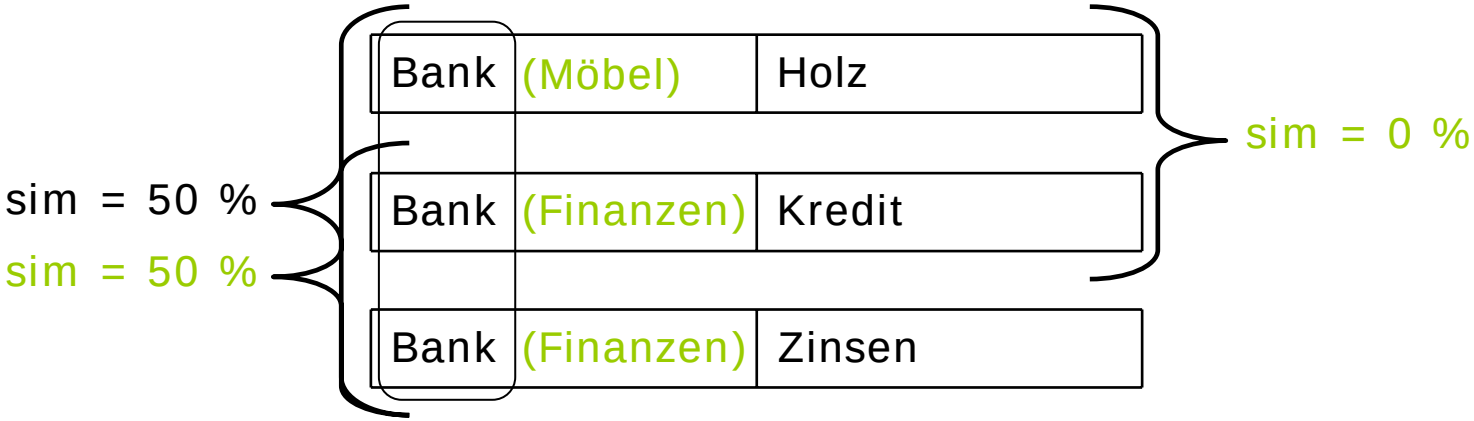
- Strukturierung und Formalisierung der Elementeigenschaften
- Stellt Elemente sowie ihre Eigenschaften in Relation

- Linguistische Datenbanken ermöglichen genauere Erfassung von Informationen aus beliebigen Texten
  - Durch Definition von Wortbedeutungen Realisierung des Konzeptmechanismus
  - Konzept ist dabei die abstrakte Darstellung einer Begrifflichkeit mit einem sinnbehafteten, eindeutigen und gemeinsamen Verständnis

Konzept basierte Recommender (1)

- Verwenden Vektorrepräsentationen mit gewichteten Termen
- Modellieren Nutzer und Elemente durch Konzepte
- Einschränkungen der CBR durch Konzepte adressierbar

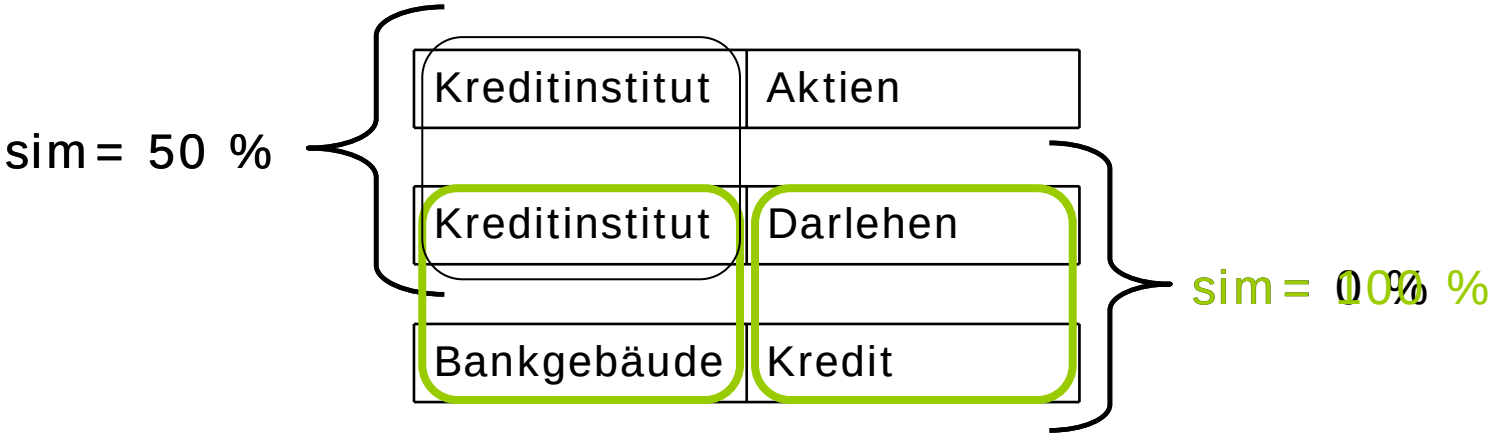
- VSM weisen Probleme mit vorhandenen Homonyme auf



- Verbessert die Precision (Genauigkeit) durch Ausschluss fehlinterpretiertbarer Terme

Konzept basierte Recommender (2)

- VSM betrachten nicht Synonyme

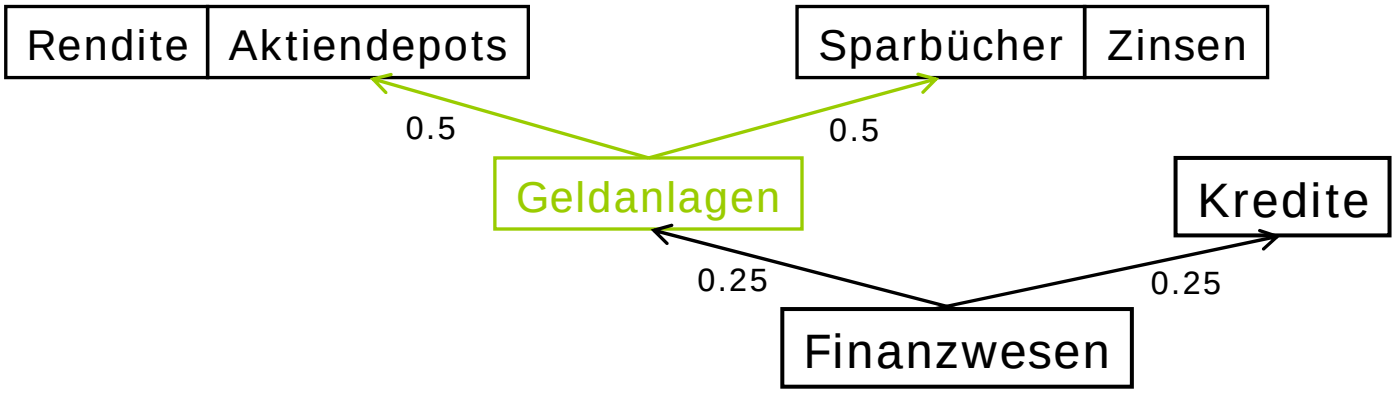


- Verbessert die Trefferquote (Recall) durch Berücksichtigung sinngemäß gleicher Worte

- Verwendung von Konzepten in den Repräsentationen steigert die Qualität der Ergebnisse

Taxonomische Recommender

- VSM berücksichtigen nicht Relationen der Konzepte
  - Orthogonalität der Vektoren bei unterschiedlichen Worten
- Erweitern Konzept-basierte Recommender durch Taxonomien
  - Hierarchisieren Konzepte durch Hypernym / Hyponym – Relation



- Profile Expansion durch Erweiterung des Vektors mit Ober / Unterbegriffen
- Dabei je nach Tiefe der Konzepte unterschiedliche Gewichte

Vorteile von Taxonomischen Recommendern

- Durch Verwendung von Konzepten Erhöhung der Qualität
  - Verbesserung um 30 % [Gonzalo et al., 1998; Navigli and Velardi, 2003]
- Verbesserte subjektive Vorhersagen durch Einbeziehung von Spezialisierungen / Generalisierungen
  - Untersuchungen ergaben je nach Domäne 40-60% [Finin et al., 2005; Bradley et al., 2000]
- Viele Probleme der „klassischen“ Recommender lösbar
  - Sparsity
  - New User
  - Overspecialization
- Anwendbarkeit auf beliebige hierarchisierbare Daten (z.B. Produktkataloge)

Erweiterungspotential durch Semantic Web Technologien

- Verbesserte Integration von Elementen durch durch die formale Definition ihrer Eigenschaften (z.B. GoodRelations)
- Eindeutige Erfassung der Vorlieben der Benutzers (z.B. GUMO)
- Cross-Site Recommendation durch gemeinsam verwendete Vokabulare
- Fehlende Daten in den Repräsentationen aus externen Quellen beziehbar (z.B. Linked Data)
- Kontextbezogene Vorschläge generierbar

Freie Universität Berlin

Vielen Dank!

http://www.corporate-semantic-web.de

INNOPROFILE
UNTERNEHMEN REGION
Die BMBF-Innovationsinitiative Neue Länder

T-Systems

STI · BERLIN