

Maschinelles Lernen II - Fortgeschrittene Verfahren

V09 Objektorientierte Probabilistisch Relationale Modelle (PRM)

Sommersemester 2017

Prof. Dr. J.M. Zöllner, Prof. Dr. R. Dillmann

INSTITUT FÜR ANGEWANDTE INFORMATIK UND FORMALE BESCHREIBUNGSVERFAHREN
INSTITUT FÜR ANTHROPOMATIK UND ROBOTIK

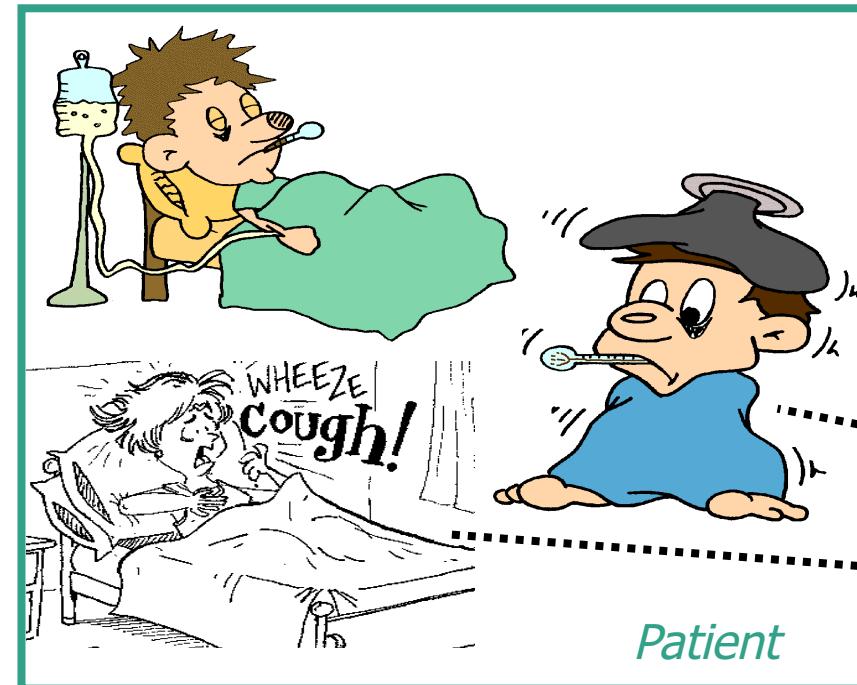


Übersicht

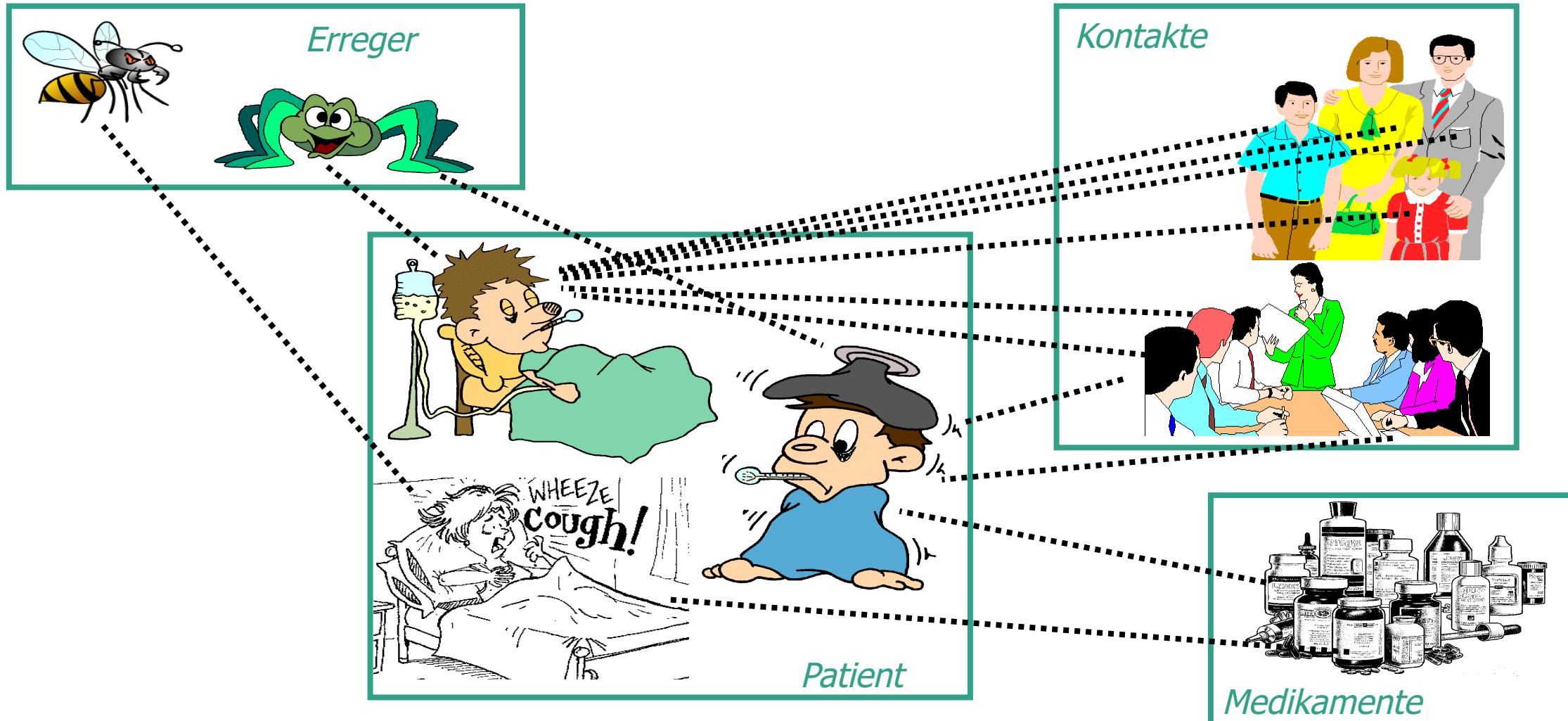
- Motivation und Grundlagen
- PRMs mit Attributunsicherheit
- PRMs mit Verknüpfungsunsicherheit
- PRMs mit Klassen-Hierarchie
- Anwendungsbeispiele

Mustererkennung in strukturierten Daten

- z.B. Herangehensweise: Symptom →(Ursache) Medizin
- Situation 2011 – siehe EHEC ← Ursache finden
- Problem: Komplexen Zusammenhang auflösen !



Mustererkennung in strukturierten Daten – komplexe Inferenz



Lernen von statistischen Modellen

■ Klassische Ansätze

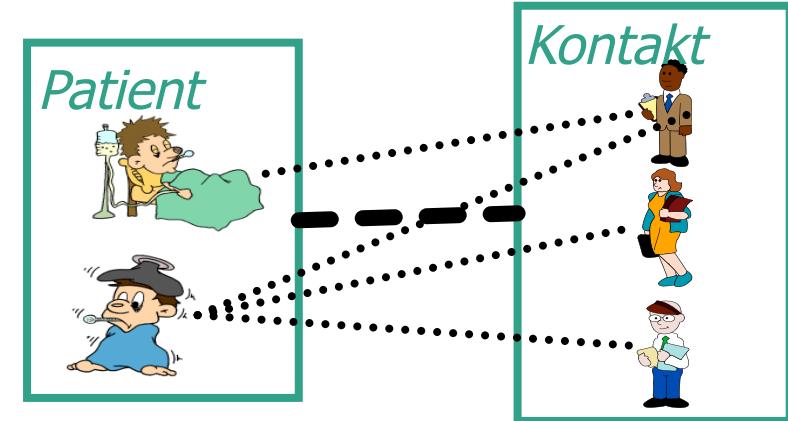
- Probabilistisch z.B. nach Bayes (unsicherheitsbehaftete Entscheidung)
- Logik - logische Schlüsse
- Funktionieren gut bei
 - flachen Strukturen (ohne weit „verzweigte“ Abhängigkeiten)
 - mit begrenzter Anzahl von Objekten
 - Annahme über Daten z.B.: unkorreliert, gleichverteilt

■ Typische Probleme:

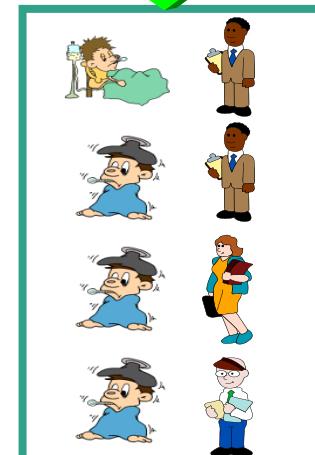
- Statistische Schiefe
- Relationale Muster können nicht erkannt werden
- Instanzen/Attribute/Prädikate müssen vordefiniert werden

■ Besser:

- Zusammenhänge modellieren, verfeinern (lernen) und probabilistische Inferenz verwenden



oft (z.B BN):
Struktur-
auflösung

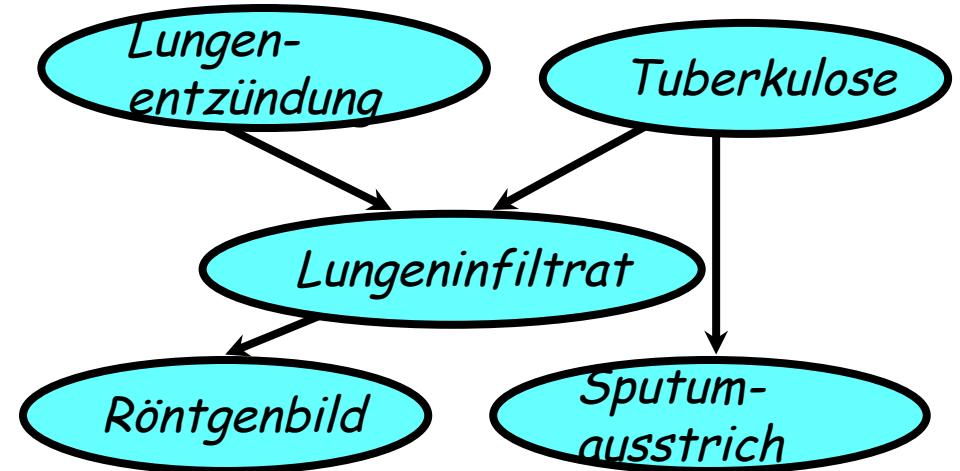


Übersicht

- Grundlagen:
 - Bayes'sche Netze (BNs)
 - Probabilistische Relationale Modelle (PRMs)
- PRMs mit Attributunsicherheit
- PRMs mit Verknüpfungsunsicherheit
- PRMs mit Klassen-Hierarchie
- Anwendungsbeispiele

Bayes'sche Netze

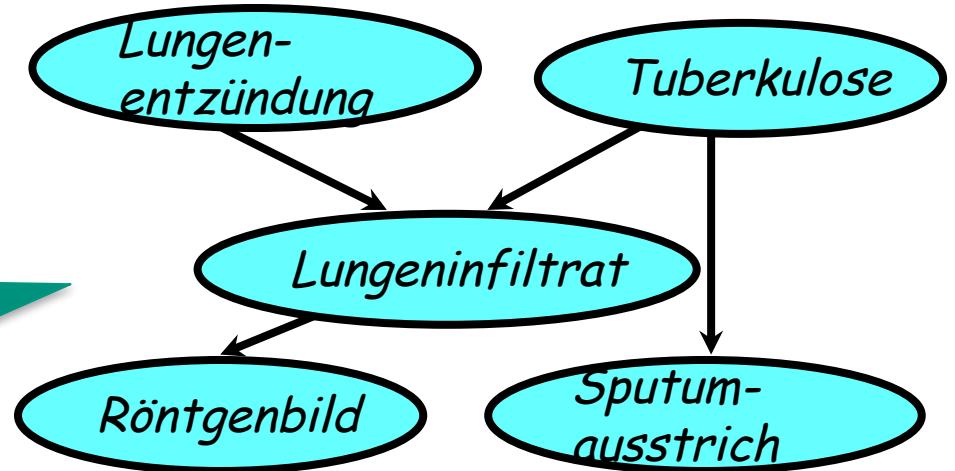
- Knoten: Zufallsvariablen
- Kanten: Bedingte Abhängigkeiten



- Graph-Struktur definiert Verbundverteilung und beinhaltet Aussagen über (bedingte) Unabhängigkeit der Zufallsvariablen, z.B.:
 - *Röntgenbild* ist (bedingt) unabhängig von *Lungenentzündung* gegeben *Lungeninfiltrat* (indirekter Effekt)

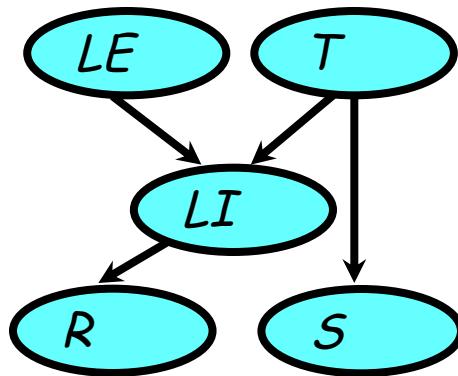
Bayes'sche Netze

le	t	$P(li le, t)$	
le	t	0,8	0,2
le	$!t$	0,6	0,4
$!le$	t	0,2	0,8
$!le$	$!t$	0,01	0,99



- Zu jedem Knoten X_i existiert eine bedingte Wahrscheinlichkeitsverteilung $P(X_i | Pa_i) =$ Verteilung über X_i für jeden Elternknoten Pa_i
 - Diskrete Variablen, P multinomial (siehe Tabelle oben)
 - P kann ebenso einer kontinuierlichen Gauß-Verteilung, Gauß-Mischverteilung, ... entsprechen

BN Semantik



Bedingte Unabhängigkeit
(in BN Struktur)

+

Lokale
Wahrscheinlich-
keitsmodelle

=

Multivariate
(Mehrdimensional)
Verbundverteilung
in Domäne

$$P(\bar{le}, t, li, r, \bar{s}) = P(\bar{le})P(t)P(li|\bar{le}, t)P(r|li)P(\bar{s}|t)$$

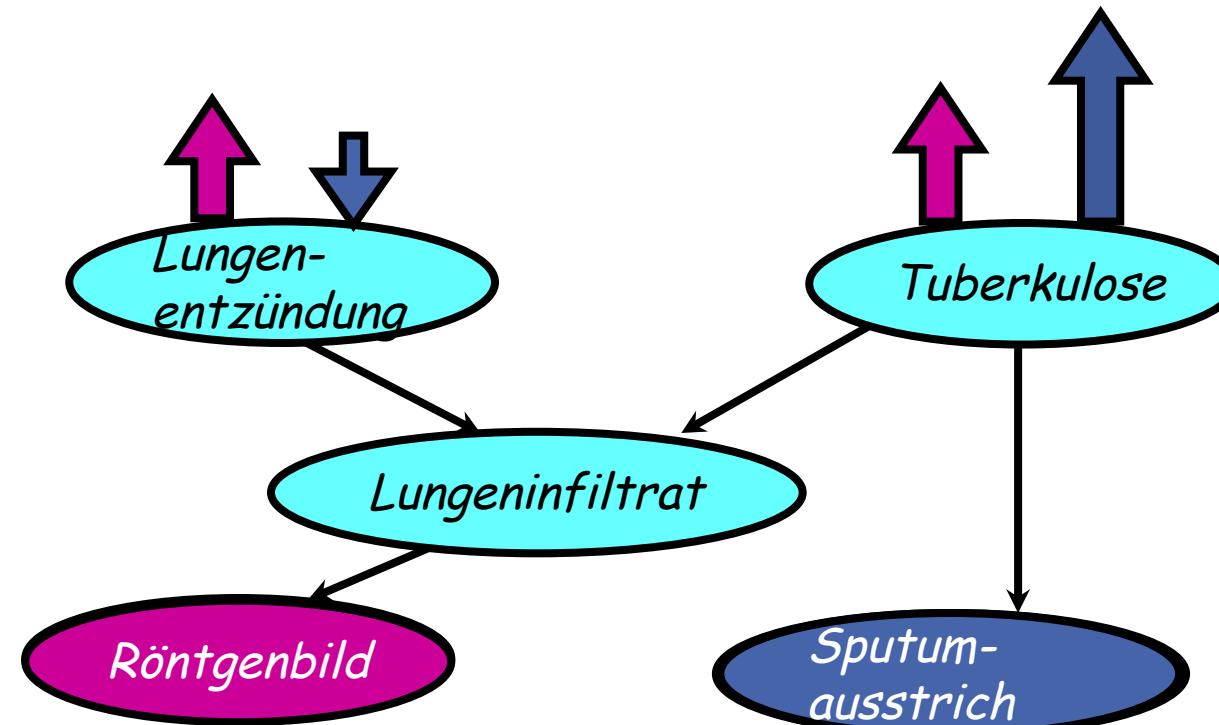
■ Wesentlicher Vorteil BN = kompakte Repräsentation:

- Gegeben
 - n Knoten (binäre Zufallsvariablen)
 - Knoten haben $\leq k$ Eltern
 - $2^k n$ vs. 2^n Parameter

Inferenz

- BN definiert über Verbundwahrscheinlichkeit Lösung zu verschiedenen Anfragen (siehe BN):

$$P(\text{Variable} \mid \text{Aussage über andere})$$



Lernen auf BN



BN Modelle können anhand empirischer Daten gelernt werden

- Vollständige Daten
 - Parameterschätzung durch numerische Optimierung
 - Strukturlernen durch kombinatorische Suche
 - NP – vollständig
- Unvollständige Daten
 - Ansätze: Expectation Maximization EM und Structural EM

Übersicht

- Grundlagen:
 - Bayes'sche Netze (BNs) → Letzte Vorlesung
 - Probabilistische Relationale Modelle (PRMs)
- PRMs mit Attributunsicherheit
- PRMs mit Struktur-/Verknüpfungsunsicherheit
- PRMs mit Klassen-Hierarchie
- Anwendungsbeispiele

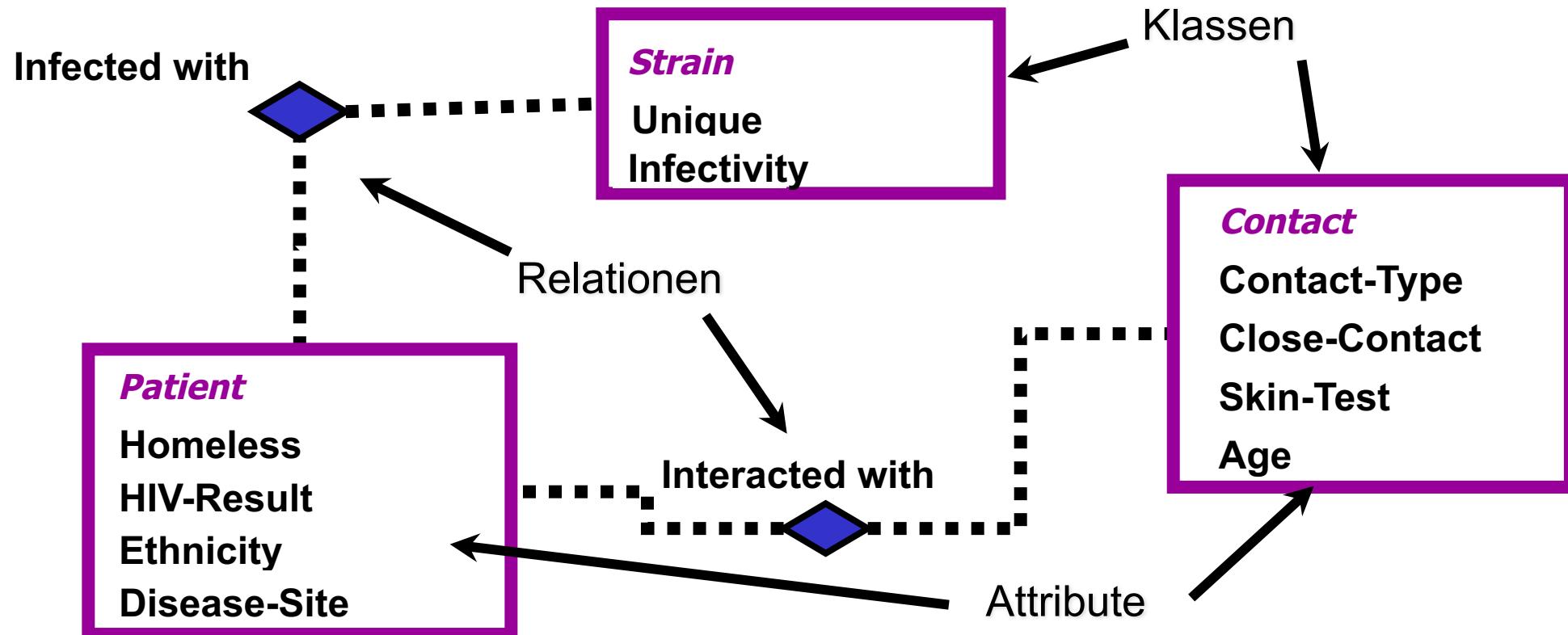
- Zusammenführen der Vorteile relationaler Logik & Bayes'scher Netze
- Vorteile der Prädikatenlogik
 - Natürliche Domänenmodellierung: Entitäten, Eigenschaften, Relationen
 - Möglichkeit zur Generalisierung über viele Instanzen derselben Domäne Schließen über Klassen von Objekte (Entitäten)
 - Erlaubt kompakte Beschreibung
- Integrieren von Unsicherheit mit relationalem Modell:
 - Kompakte, natürliche probabilistische Modelle
 - Eigenschaften von Entitäten der Domäne können auf Eigenschaften von anderen Entitäten beruhen
 - Unsicherheit über relationale Struktur der Domäne

Definition: Entität, Attribut, Instanz

- Klassenmenge $X = \{X_1, \dots, X_n\}$
 - auch Menge der Entitäten genannt
- Für jede Klasse X :
 - Menge von Attributen $X.A$
 - Wertebereich $V(X.A)$ für jedes Attribut
 - Attribute können Zufallsvariablen sein
- Eine Instanz I (der Welt) definiert für jede Klasse X die Objekte der Klasse $I(X) = \{o_1, \dots, o_n\}$ (die zu betrachten sind)

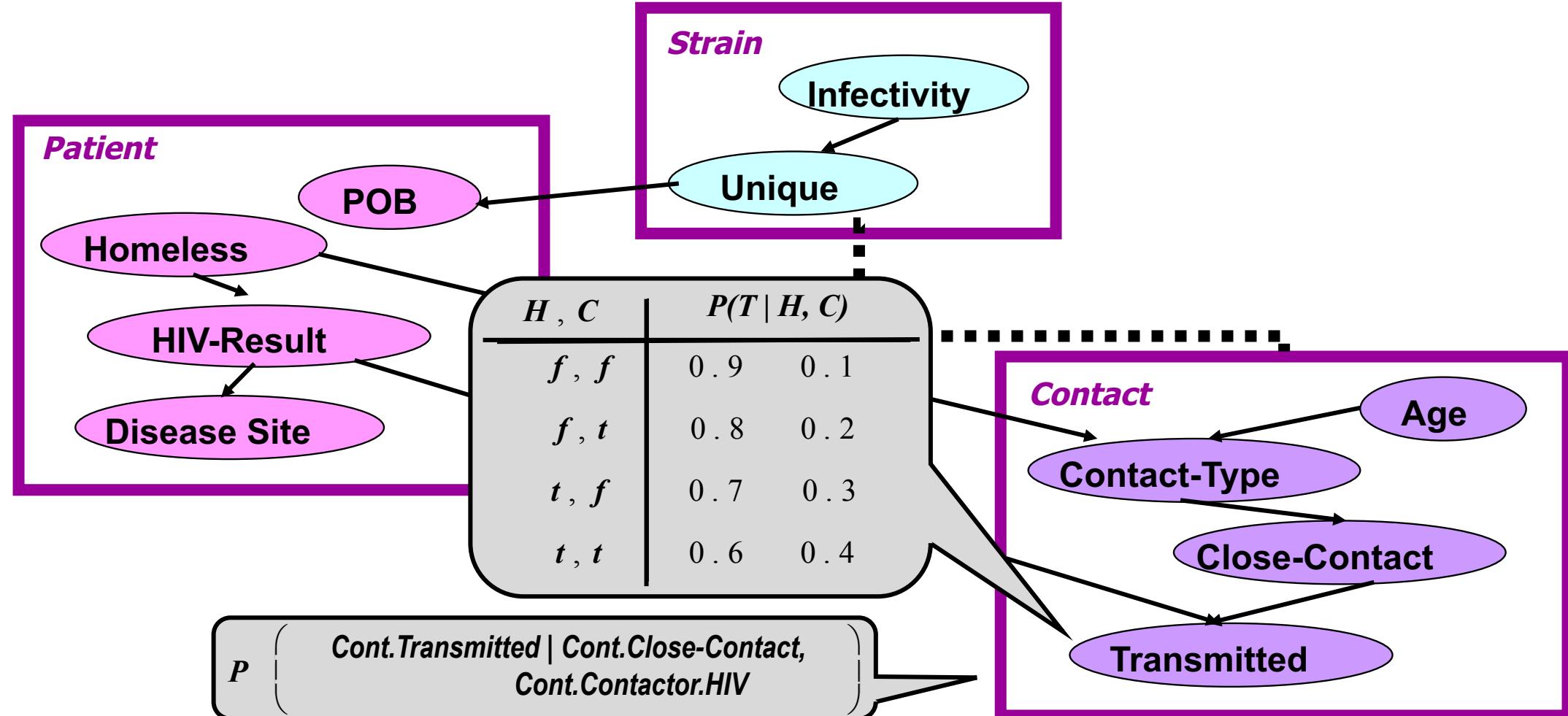
Definition: Relationales Schema

- Beschreibt Klassen von Objekten (u. Attribute) und Relationen
- Entspricht der Prädikatenlogik



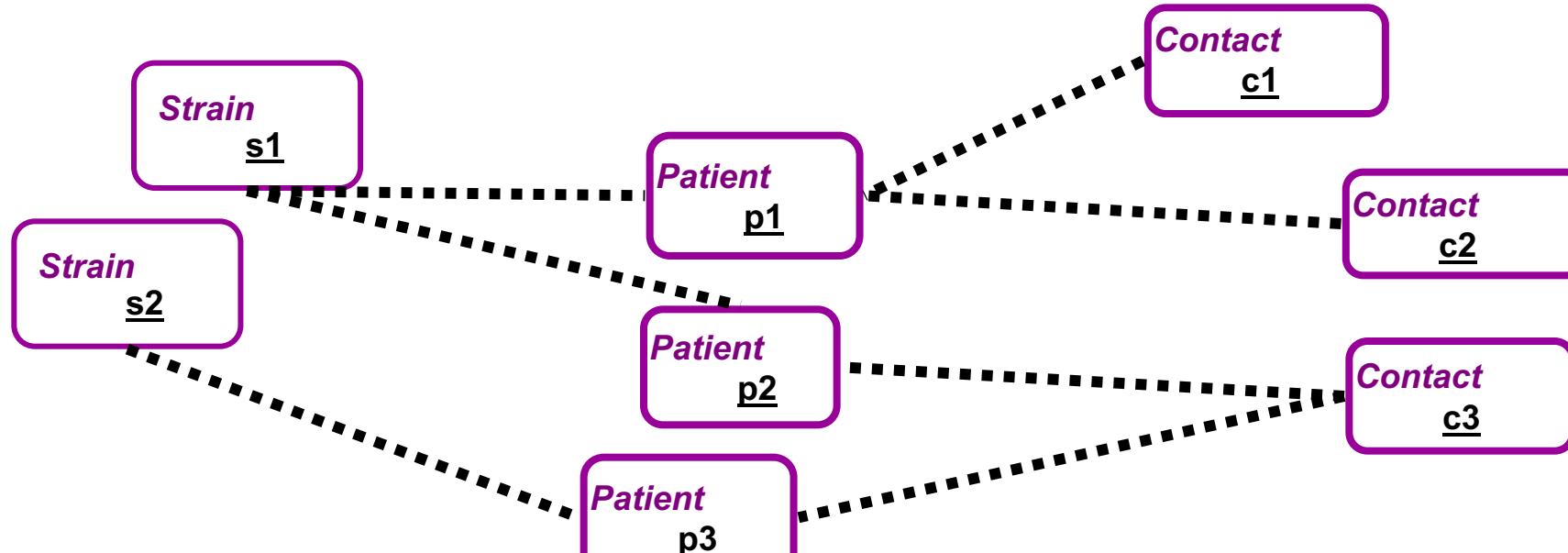
Probabilistisches Relationales Modell

- Beschreibung der Attribute z.B. als Zufallsvariablen und bedingte Abhängigkeiten im relationalen Schema

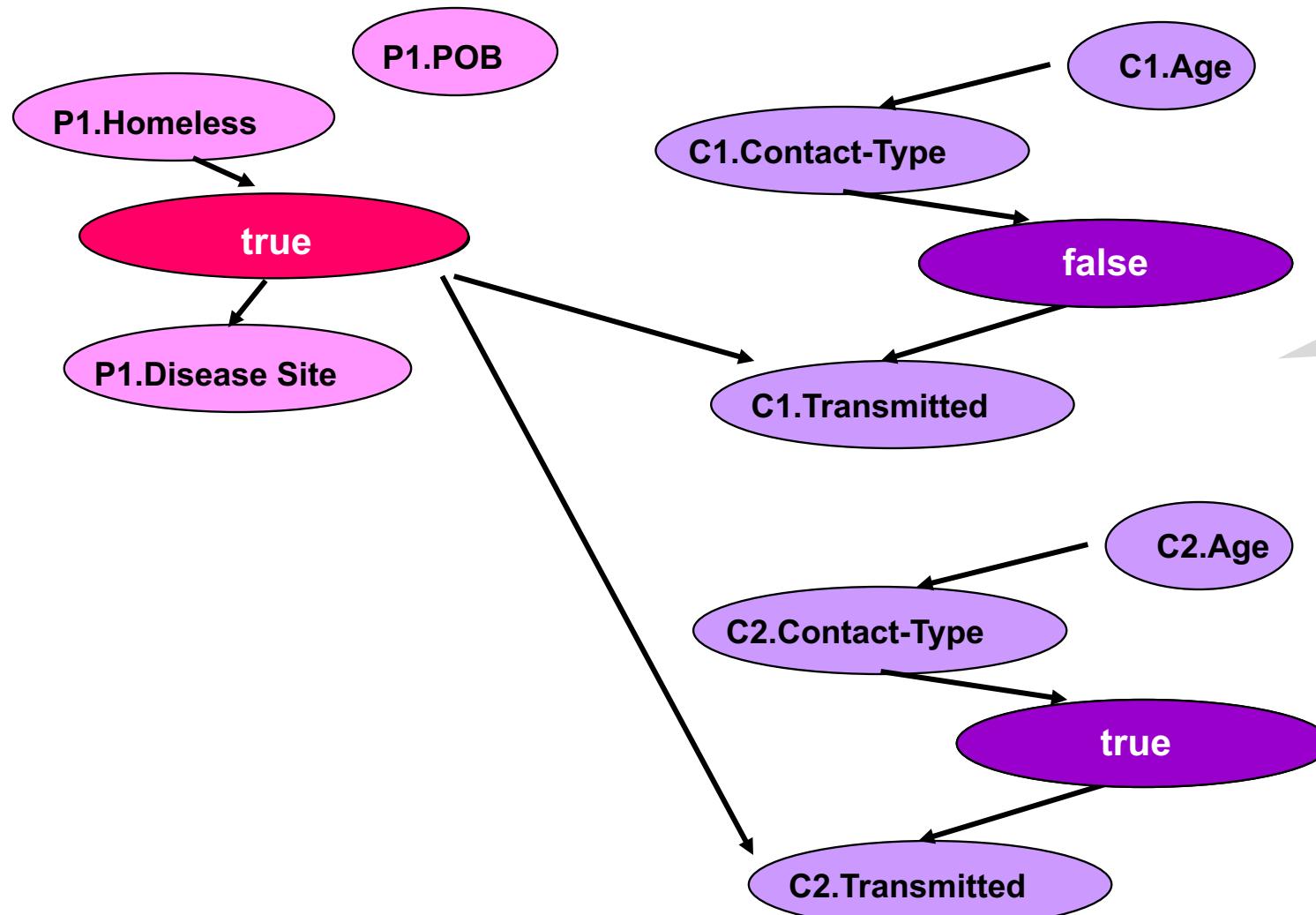


Relationales Skelett: Instanziierung

- Das relationale Skelett σ definiert für eine Instanz
 - Menge an Objekten jeder Klasse
 - Abhängigkeiten zwischen diesen
- Unsicherheit über Attribute (Zufallsvariablen)
- PRM definiert für entsprechende σ die Wahrscheinlichkeitsverteilungen (über alle Instanzen) der Attribute



PRM + Skelett = Äquivalenz zu einem entsprechenden Bayes'schen Netz



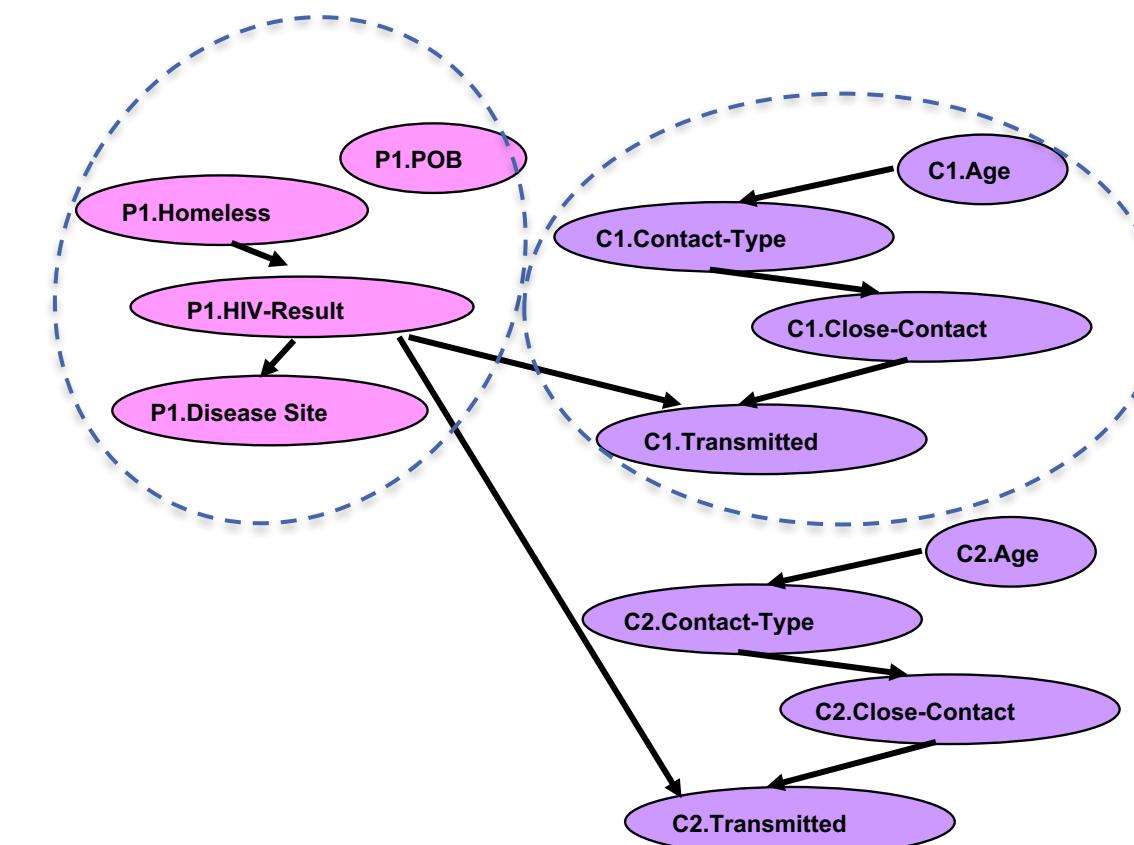
H, C	$P(T H, C)$	
f, f	0.9	0.1
f, t	0.8	0.2
t, f	0.7	0.3
t, t	0.6	0.4

H, C	$P(T H, C)$	
f, f	0.9	0.1
f, t	0.8	0.2
t, f	0.7	0.3
t, t	0.6	0.4

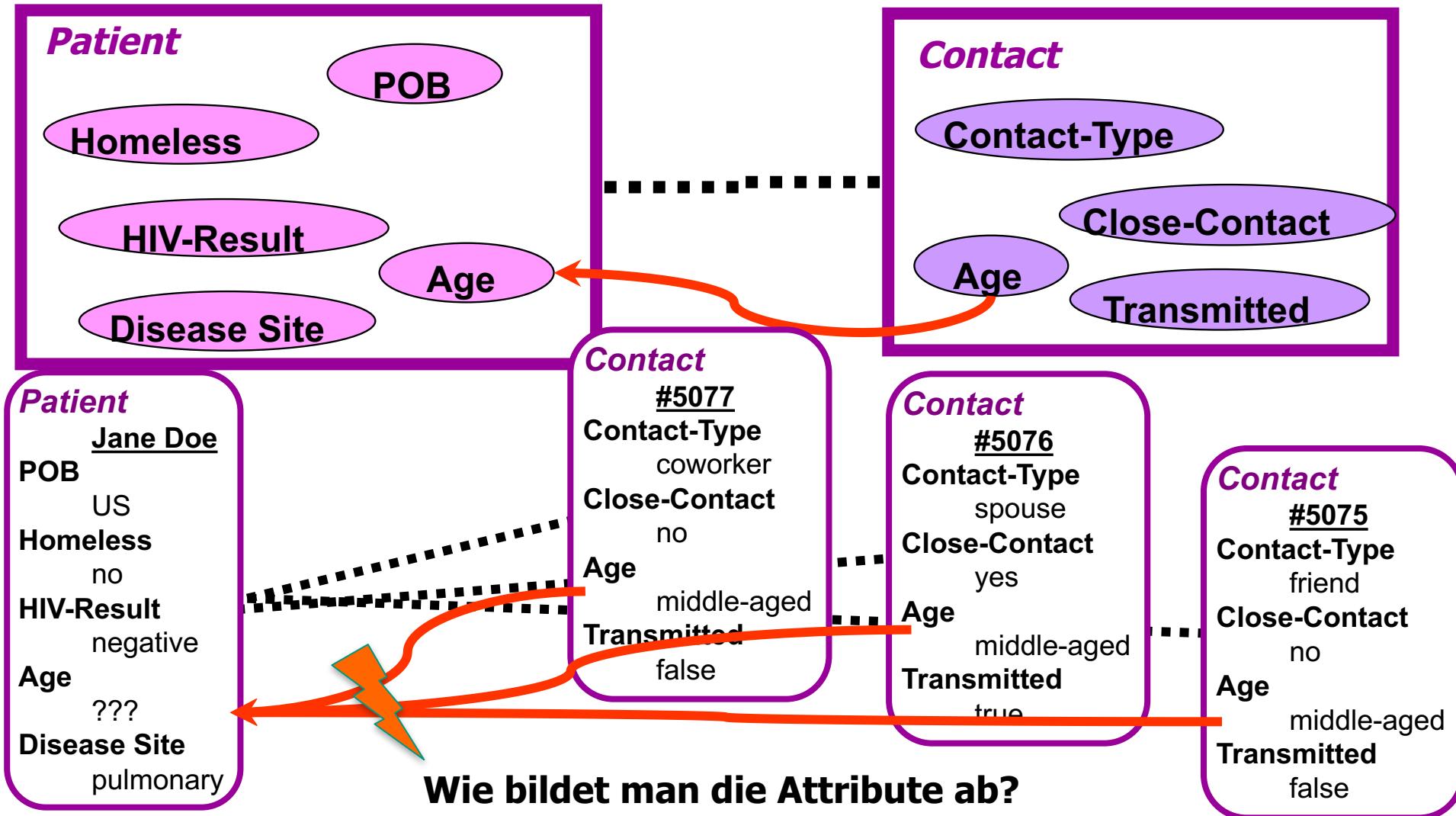
PRM Inferenz

- Über konstruiertes, äquivalentes BN (\leftarrow PRM+Skelett)
 → kann sehr groß werden, d.h. nicht effizient, Struktur wird vernachlässigt

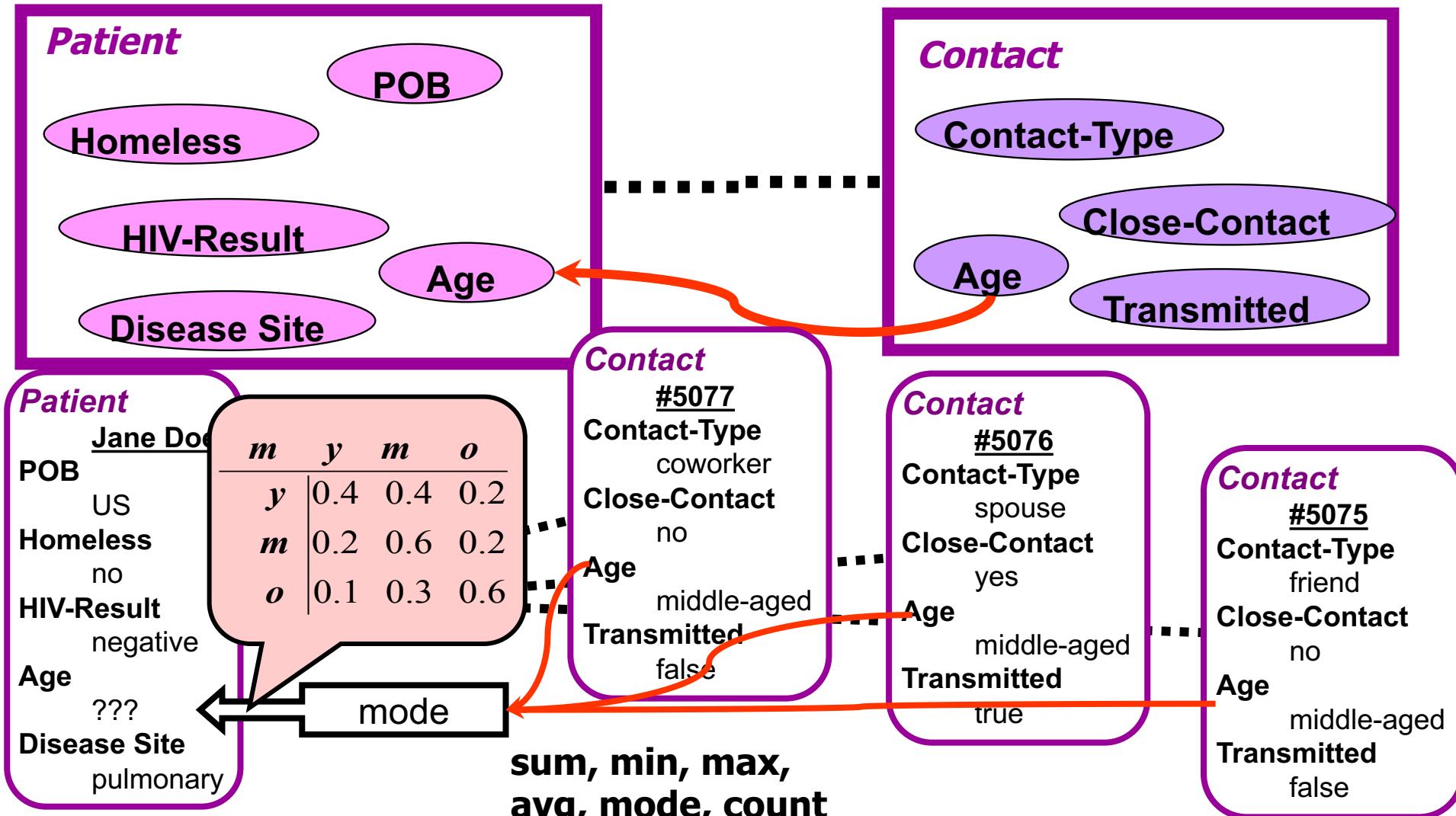
- Besser:
 - Methode: Structured Variable Elimination (SVE)
 - Nutzen der Kapselung und Wiederverwendung
 - Strukturmodell gibt die „Kapselung“
 - Wiederverwendung der inferierten Teilgraphen wenn Evidenzen gleich bleiben



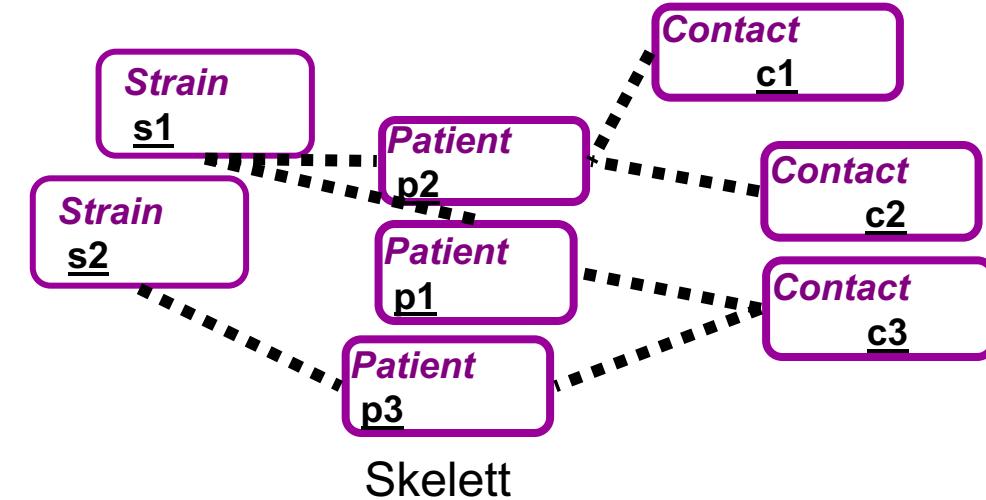
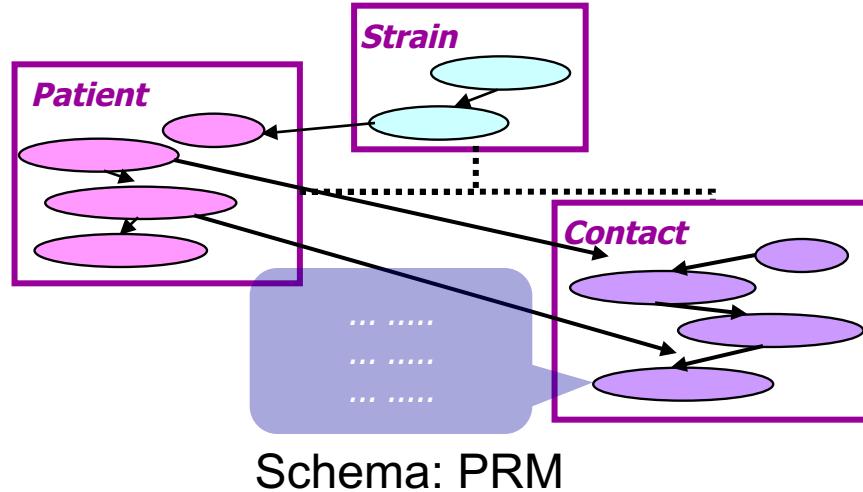
PRM mit mehreren Objekt - Instanzen → Abhängigkeiten aggregieren



PRM mit mehreren Objekt - Instanzen → Abhängigkeiten aggregieren



Semantik von PRM mit Attributunsicherheit (AU)



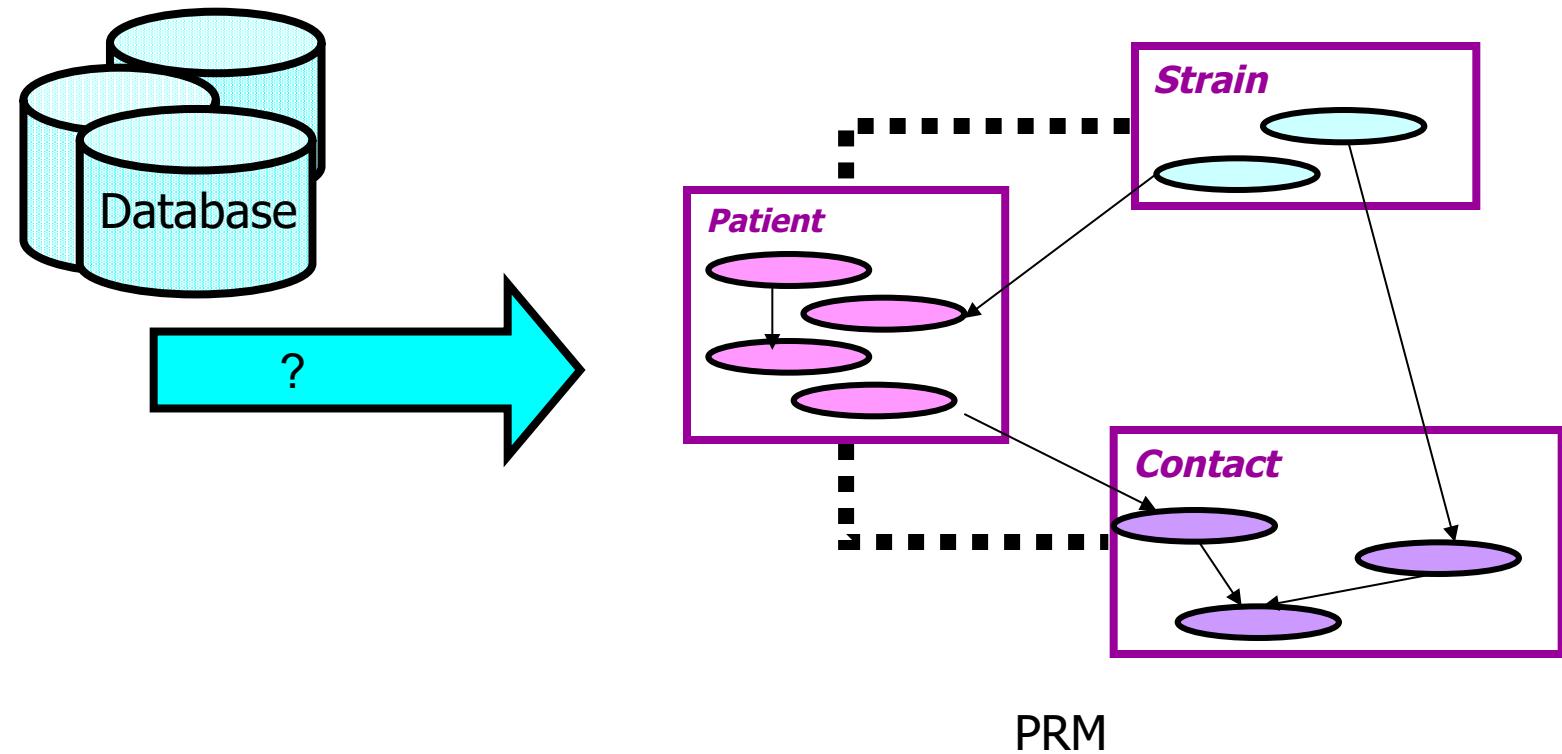
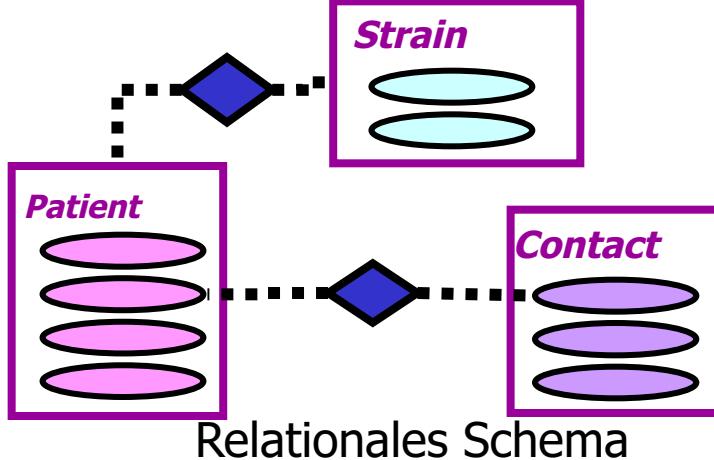
■ Wahrscheinlichkeit einer Instanzen I (der Welt):

$$P(I | \sigma, S, \Theta) = \prod_{x \in \sigma} \prod_{x.A} P(x.A | \text{parents}_{S, \sigma}(x.A))$$

↑
Objekte ↑
Attribute

Lernen von PRMs mit AU

- Verwenden von Lerndaten um
 - Parameter (i.A. lokale Wahrscheinlichkeiten) zu schätzen
 - Struktur zu bestimmen
- Wie?



Parameterschätzung in PRMs

- Annahme, dass Abhängigkeitsstruktur S bekannt ist und Daten vollständig beobachtbar sind
- Ziel: Schätzung der PRM Parameter θ :

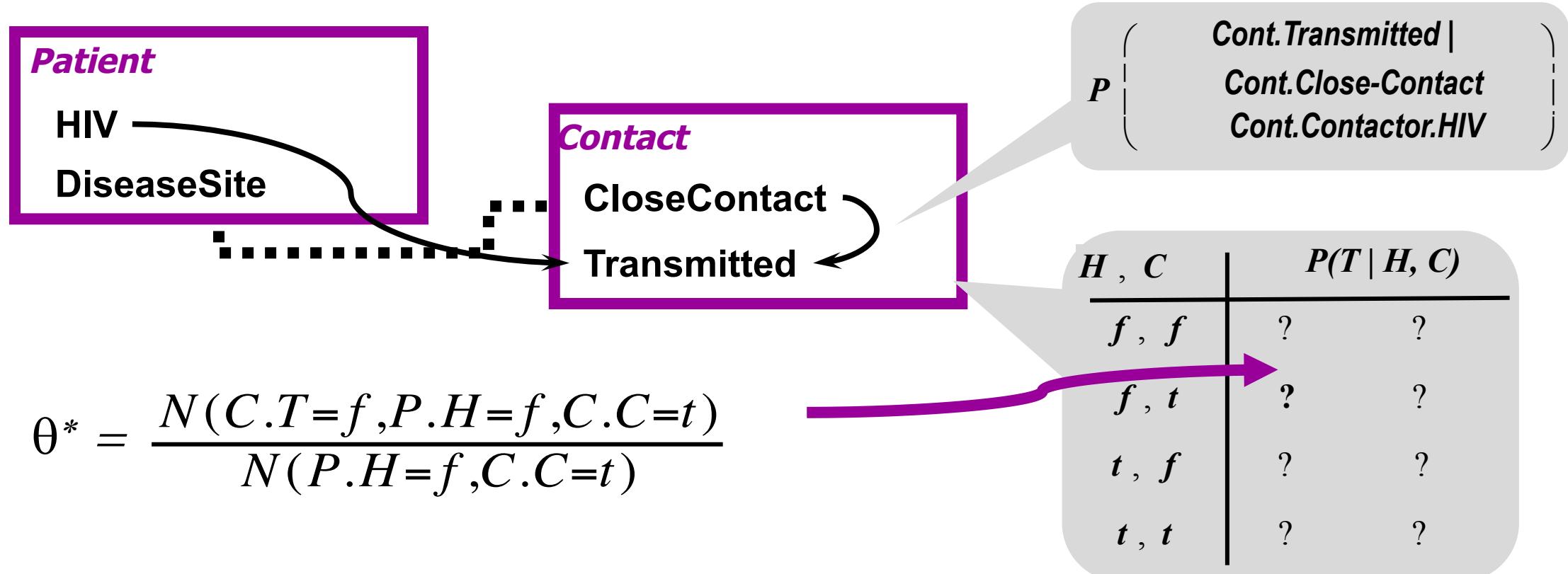
$$\theta_{x.A|parents(x.A)}$$

- z.B. im diskreten Fall: θ = Einträge in lokalem Wahrscheinlichkeitsmodell
- θ ist gut, wenn es wahrscheinlich ist, dass die beobachteten Daten der Instanz I erzeugt werden

$$l(\theta : I, S) = \log P(I | S, \theta)$$

- MLE Prinzip: Wähle θ^* so, dass l maximal ist

MLE Parameterschätzung (diskret)



$$\theta^* = \frac{N(C.T=f, P.H=f, C.C=t)}{N(P.H=f, C.C=t)}$$

- Vollständige Daten: Schätzung basierend auf Häufigkeiten in den Lerndaten

MLE Parameterschätzung



$$P \left(\begin{array}{l} \text{Cont. Transmitted} \\ | \\ \text{Cont. Close-Contact} \\ | \\ \text{Cont. Contactor.HIV} \end{array} \right)$$

H, C	$P(T H, C)$	
f, f	?	?
f, t	?	?
t, f	?	?
t, t	?	?

θ instanziieren

$$\begin{aligned}
 E[N(C.T = f, P.H = f, C.C = t)] \\
 &= \sum_{\text{instanzen } i} P(C.T^i = f, \dots | \text{evidenzen}^i) \\
 \theta &= \frac{E[N(C.T = f, P.H = f, C.C = t)]}{E[N(P.H = f, C.C = t)]}
 \end{aligned}$$

- Unvollständige Daten: EM (benötigt Inferenz !!)

Strukturbestimmung

■ Idee:

- Verwenden einer Bewertungsfunktion für PRM - Strukturen
- Lokale Suche über gültige Strukturen

■ Wesentliche Schritte:

- Bestimmung gültiger Modelle
- Definition der Bewertungsfunktion
- Suche im Modellraum

Strukturbestimmung

■ Idee:

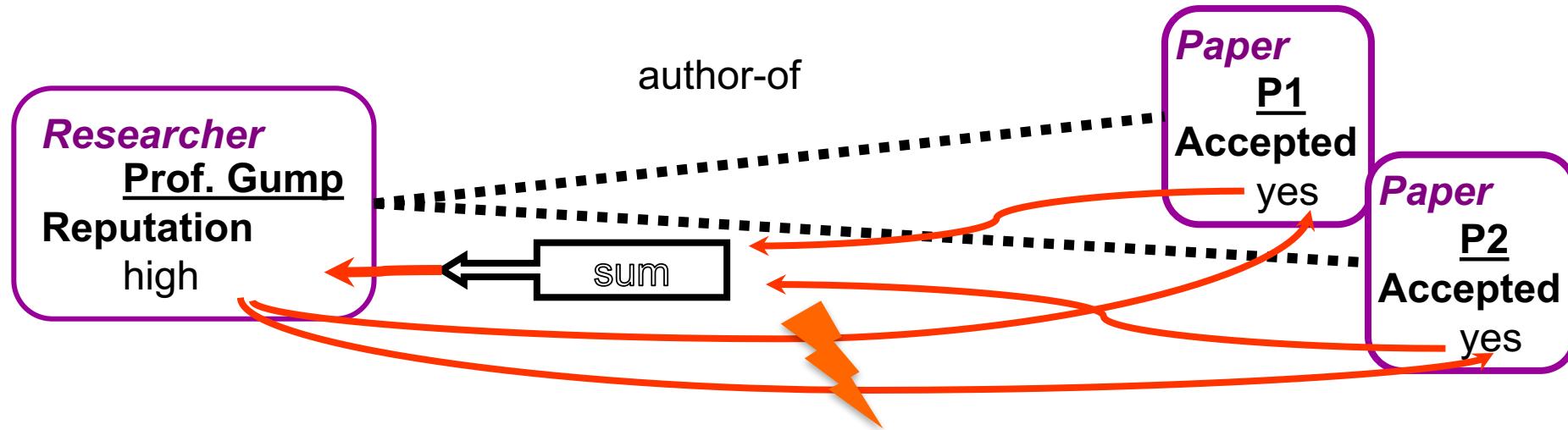
- Verwenden einer Bewertungsfunktion für PRM - Strukturen
- Lokale Suche über gültige Strukturen

■ Wesentliche Schritte:

- Bestimmung gültiger Modelle
- Definition der Bewertungsfunktion
- Suche im Modellraum

Bestimmung gültiger Modelle

- PRM definieren ein kohärentes Wahrscheinlichkeitsmodell über ein Skelett σ , wenn die Abhängigkeiten zwischen Objektattributen azyklisch sind



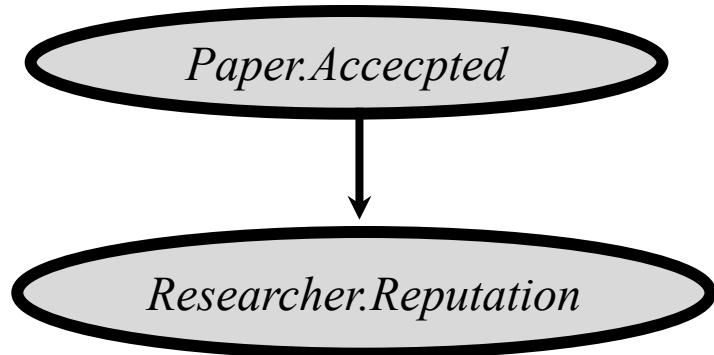
Wie garantieren wir, dass ein PRM für **jedes** Skelett azyklisch ist?

Attribut-Abhängigkeit

PRM
Abhängigkeitsmodell



Abhängigkeitsgraph
für Attribute



Kante wenn:
Attribut *Researcher.Reputation*
direkt abhängig ist von Attribut
Paper.Accepted

Es gilt:

Azyklisch für Klassenabhängigkeiten → Zyklenfreiheit für alle Instanzen
Azyklischer Abhängigkeitsgraph → azyklisch für jedes σ (Instanziierung)

(Erweiterungen möglich um bestimmte Zyklen zu erlauben)

Strukturbestimmung

■ Idee:

- Verwenden einer Bewertungsfunktion für PRM - Strukturen
- Lokale Suche über gültige Strukturen

■ Wesentliche Schritte:

- Bestimmung gültiger Modelle
- **Definition der Bewertungsfunktion**
- Suche im Modellraum

Bewertungsfunktion /-methode

- Bewertung einer Struktur S gegeben eine Instanz I
- Log-Likelihood + Bayes'scher Ansatz:

$$Score(S : I) = \log P(S|I) \propto \log \underbrace{P(I|S)}_{\text{marginal likelihood}} \underbrace{P(S)}_{\text{prior}}$$

- Standard-Ansatz zur Bewertung von Modellen

Strukturbestimmung

■ Idee:

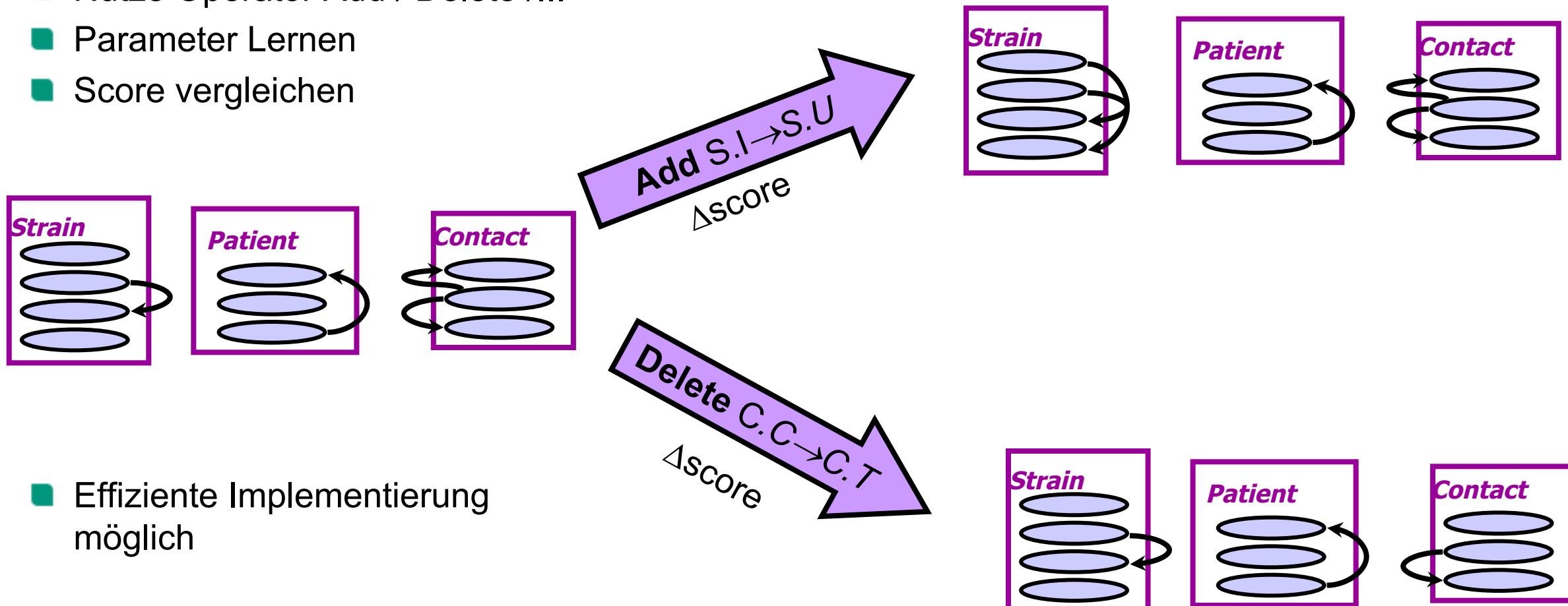
- Verwenden einer Bewertungsfunktion für PRM - Strukturen
- Lokale Suche über gültige Strukturen

■ Wesentliche Schritte:

- Bestimmung gültiger Modelle
- Definition der Bewertungsfunktion
- **Suche im Modellraum**

Suche im Modellraum

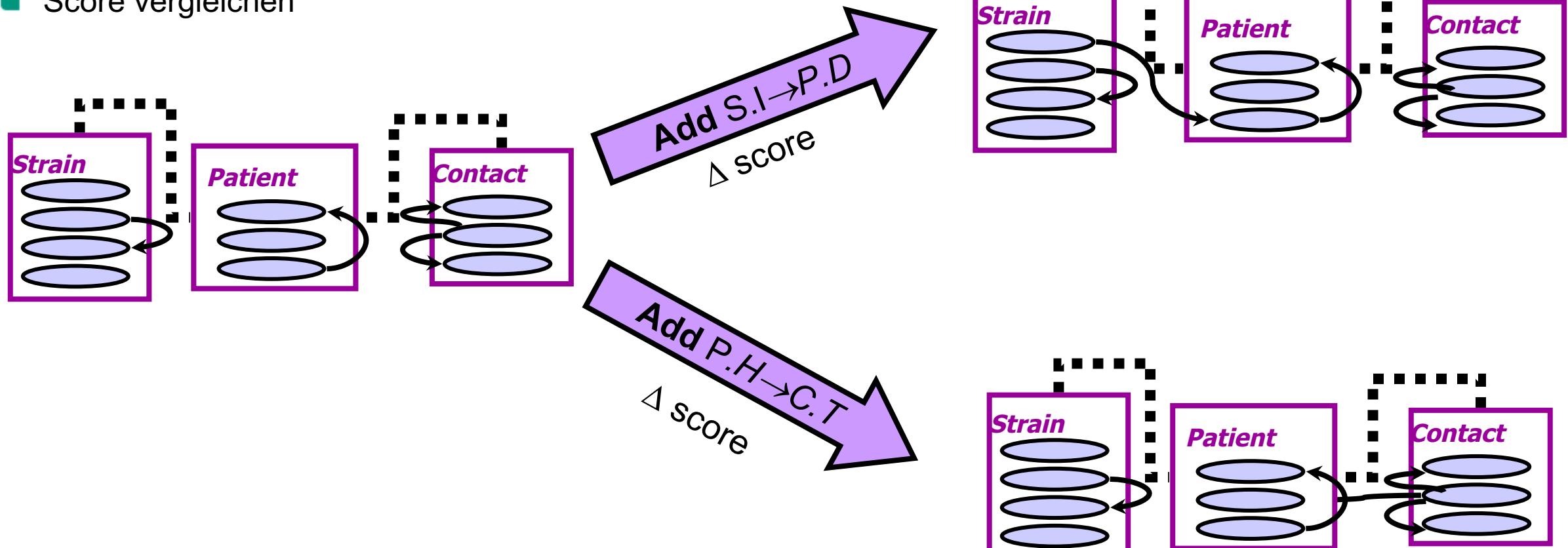
- Phase 0: Betrachte nur Abhängigkeiten innerhalb einer Klasse
- Nutze Operator Add / Delete /...
- Parameter Lernen
- Score vergleichen



- Effiziente Implementierung möglich

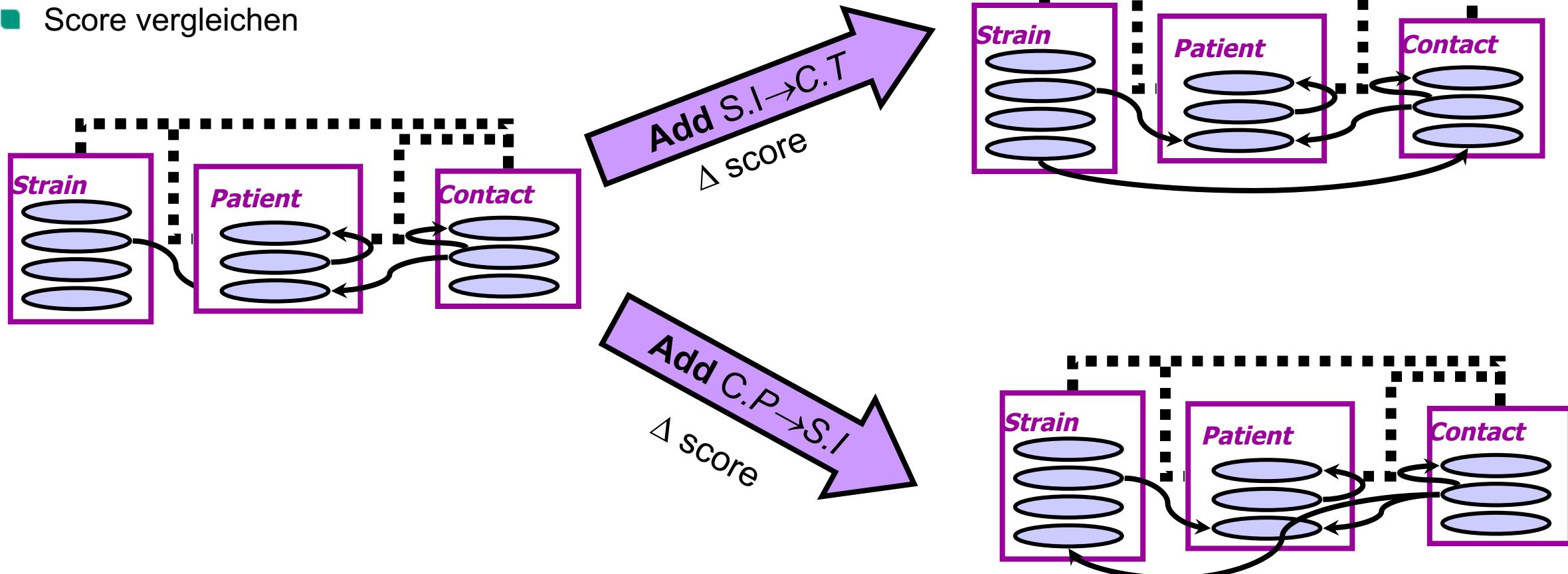
Suche im Modellraum

- Phase 1: Betrachte nur Abhängigkeiten benachbarter Klassen, anhand des relationalen Schema
- Operator: Add/...
- Parameter Lernen
- Score vergleichen



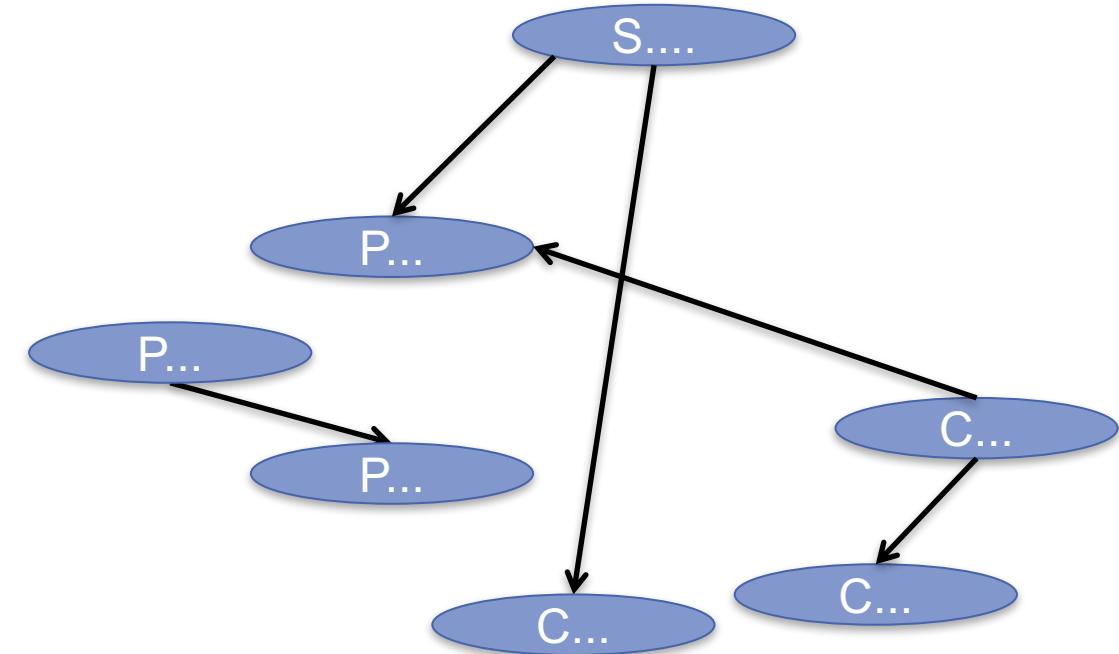
Suche im Modellraum

- Phase 2: Betrachte Abhängigkeiten weiterer Klassen, anhand relationaler Abhängigkeitsketten
- Operator:
- Parameter Lernen
- Score vergleichen



Suche im Modellraum

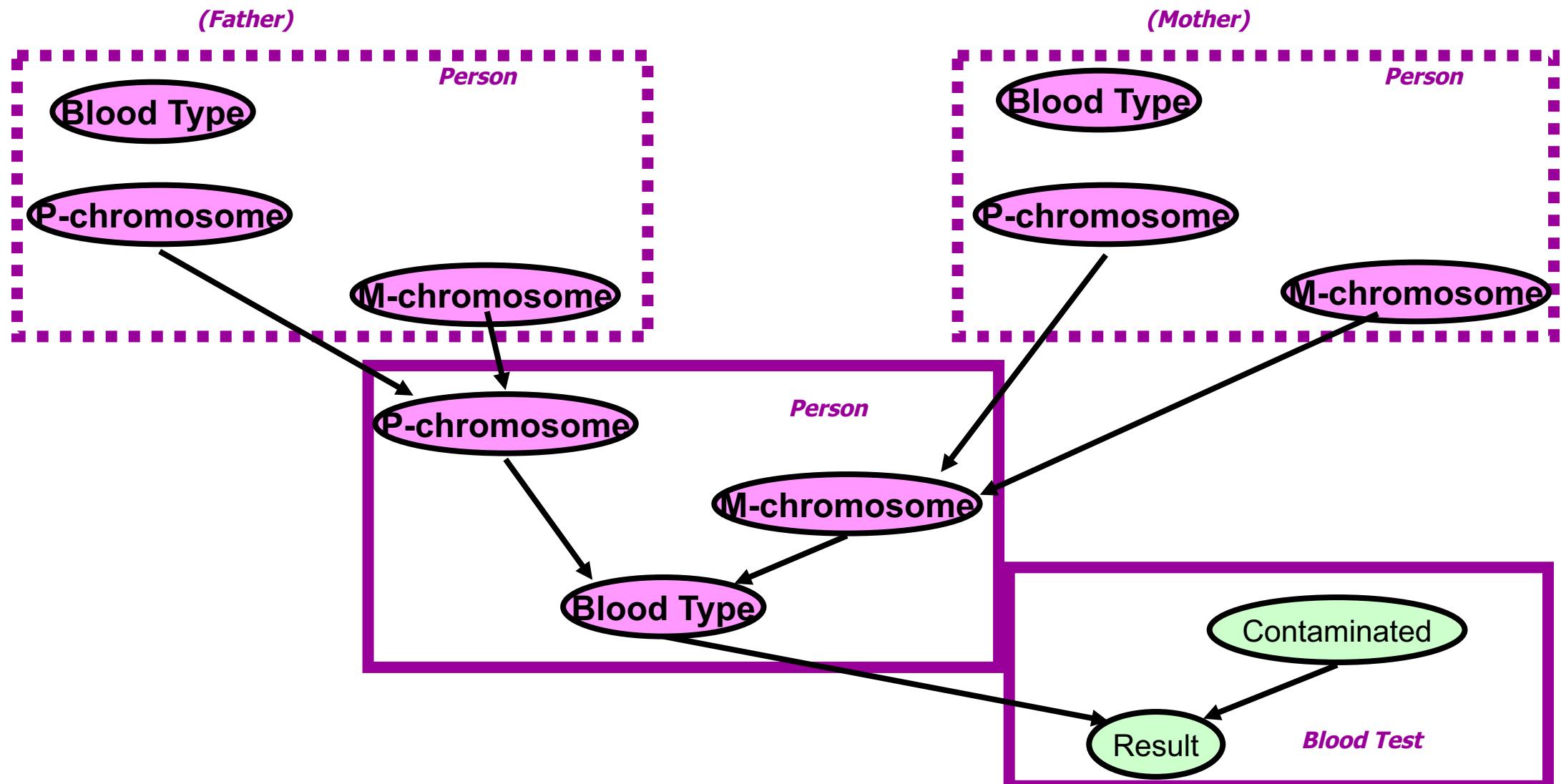
- Wichtig: Lokale Suche nur im Raum der zulässigen Strukturen
- Bei Übergangsoperationen zwischen Strukturen (*add edge, delete edge, reverse edge, ...*)
 → Abhängigkeitsgraphen – Konsistenzprüfung
 - Nach Hinzufügen einer Kante (u,v)
 prüfen ob ein Weg von v nach u existiert
 - Komplexität der Tiefensuche
 $O(|\text{Knoten}| + |\text{Kanten}|)$
- Lernen bei BN bereits
 NP – schwer,
 PRM sind noch komplexer



Beispiel 1 -Einfache „Genetik“ Domäne

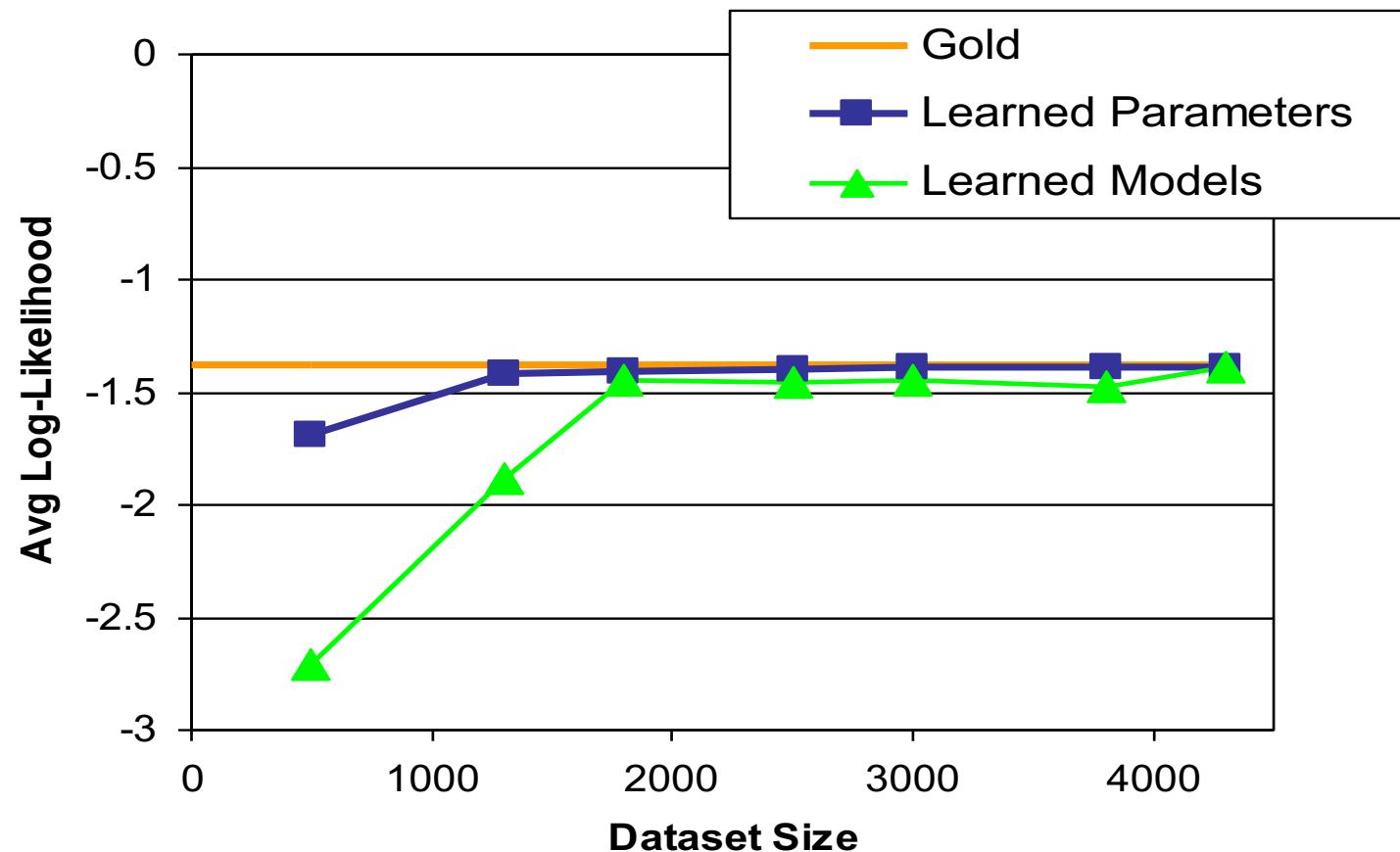
- Synthetische Problem: Blut-Gruppe bestimmen Abhängigkeiten
 - Vererbung eines Gen, das die Blut-Gruppe bestimmt
 - Jede Person 2 Kopien der Chromosomen, die Gene enthalten (von Mutter und Vater)
 - Bluttest (zur Bestimmung)
 - Kontaminierte Blutproben sind möglich
- Lernen auf Trainingsdatensätze verschiedener Größen
- Evaluierung: Vergleich der log-likelihood Ergebnisse eines Testdatensatzes der Größe 100.000 Fälle
 - „Gold“ – Standardmodell der Genetik (ground-truth)
 - PRM mit Lernen von Parameter (Modellstruktur gegeben)
 - PRM mit Lernen des Modells (Lernen sowohl der Struktur als auch der Parameter)

Beispiel 1: Korrektes PRM



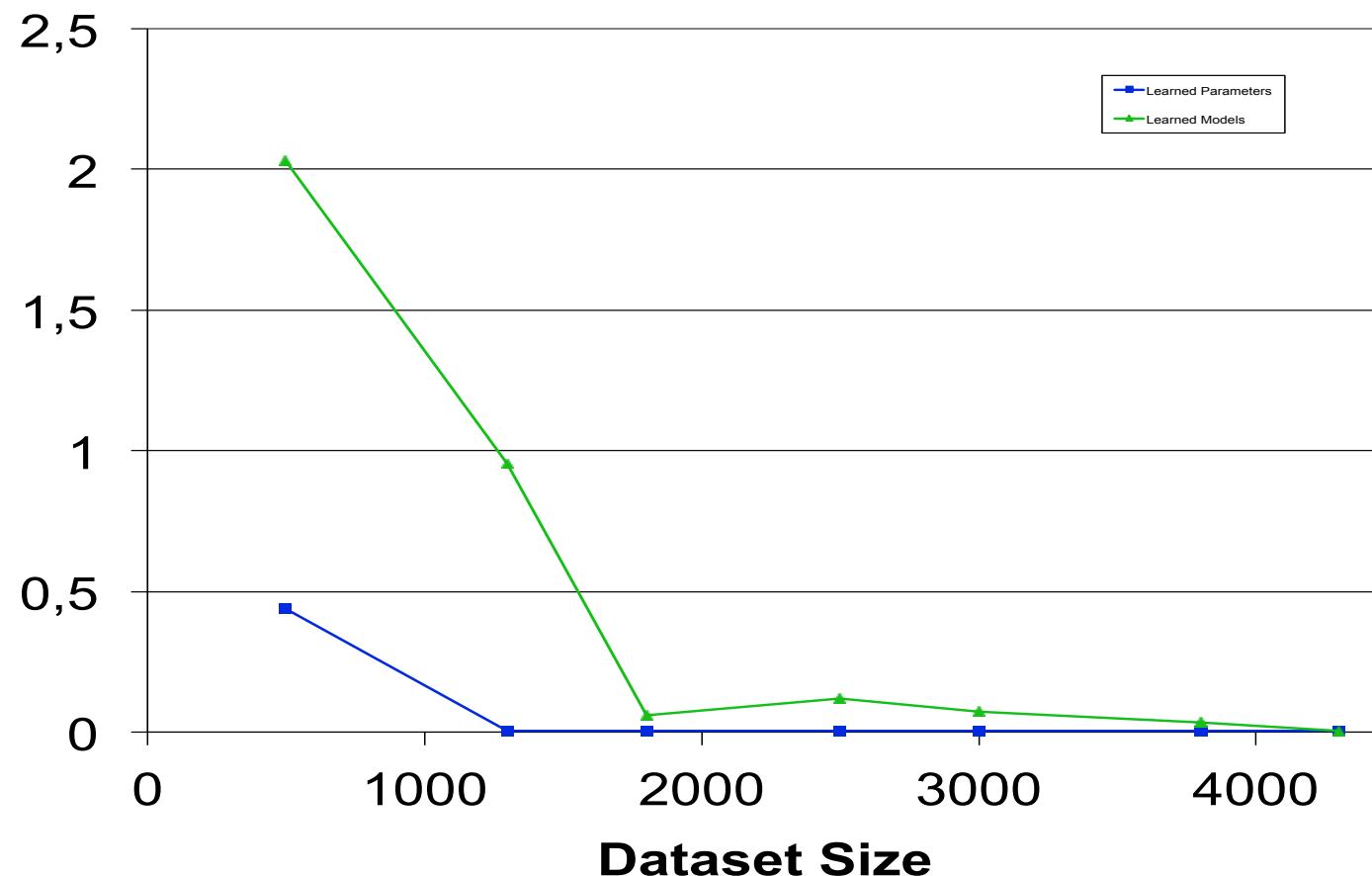
Beispiel 1: Fehler am Testdatensatz

- Vergleichbare Ergebnisse aber mehr Lerndaten für Lernen der gesamten Struktur nötig ~ 4000 Lerndaten



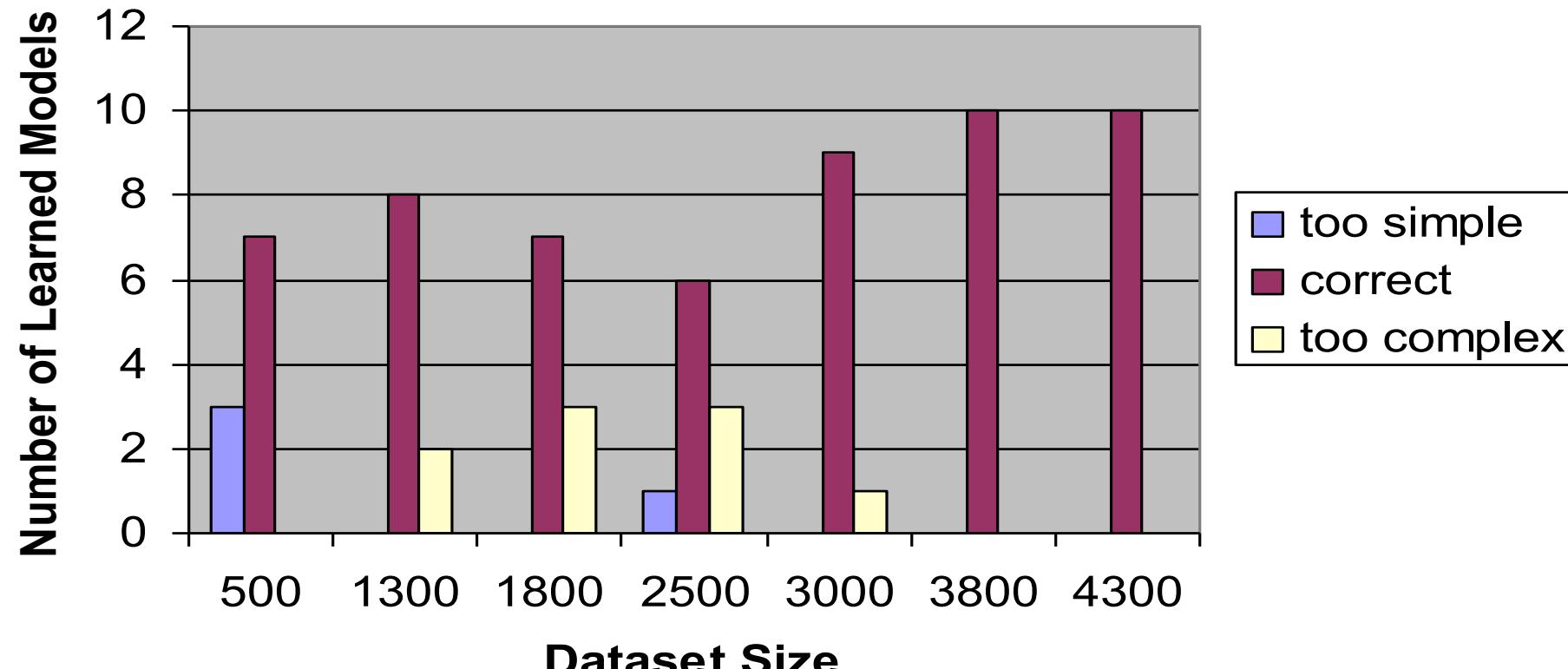
Beispiel 1: Fehler Varianz

- Konvergiert langsam



Beispiel 1: Fehler - Strukturlernen

- Vergleich generierter Modelle
 - Lernen mit unterschiedlich vielen Daten
 - jeweils eine begrenzte Anzahl von 10 Modellen



Beispiel 2: Reale Tuberkulose (TB)

Patient (≈ 2300)

Ethnicity

Homeless

Age @ diagnosis

HIV result

Disease-site

X-ray

⋮
⋮

Contact (≈ 20000)

Contact-type

Age

Care

Infected

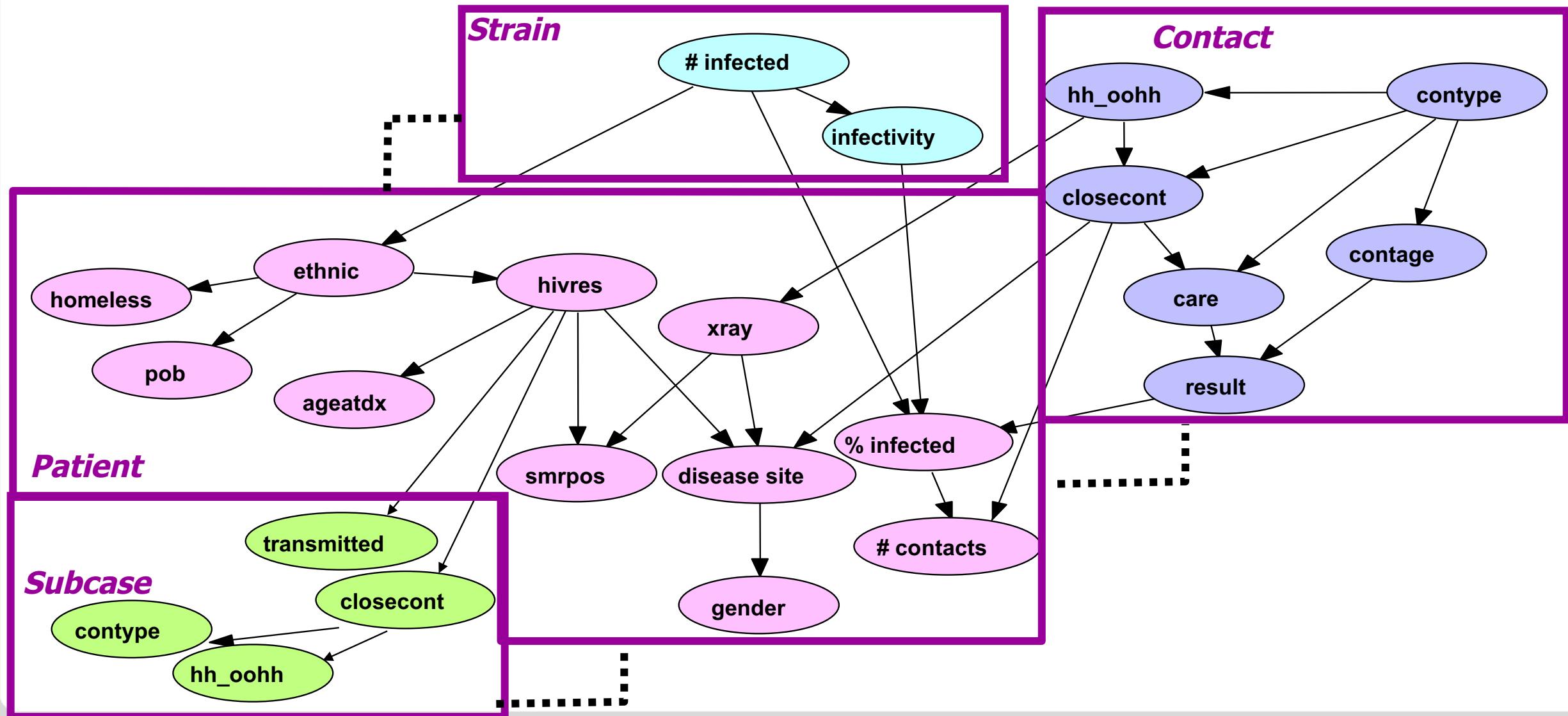
⋮
⋮

Strain (≈ 1000)

Unique

Drug-Resistance

Beispiel 2: TB Erzeugte Struktur (schematisch)



Übersicht

- Motivation und Grundlagen
- PRMs mit Attributunsicherheit
- PRMs mit Struktur-/ Verknüpfungsunsicherheit
 - Existiert eine Verknüpfung zw. Entitäten → Welche ?
 - Literatur !
- PRMs mit Klassen-Hierarchie
- Anwendungsbeispiele

Übersicht

- Motivation und Grundlagen
- PRMs mit Attributunsicherheit
- PRMs mit Struktur-/ Verknüpfungsunsicherheit
- PRMs mit Klassen-Hierarchie
- Anwendungsbeispiele

Motivation - Limitierungen

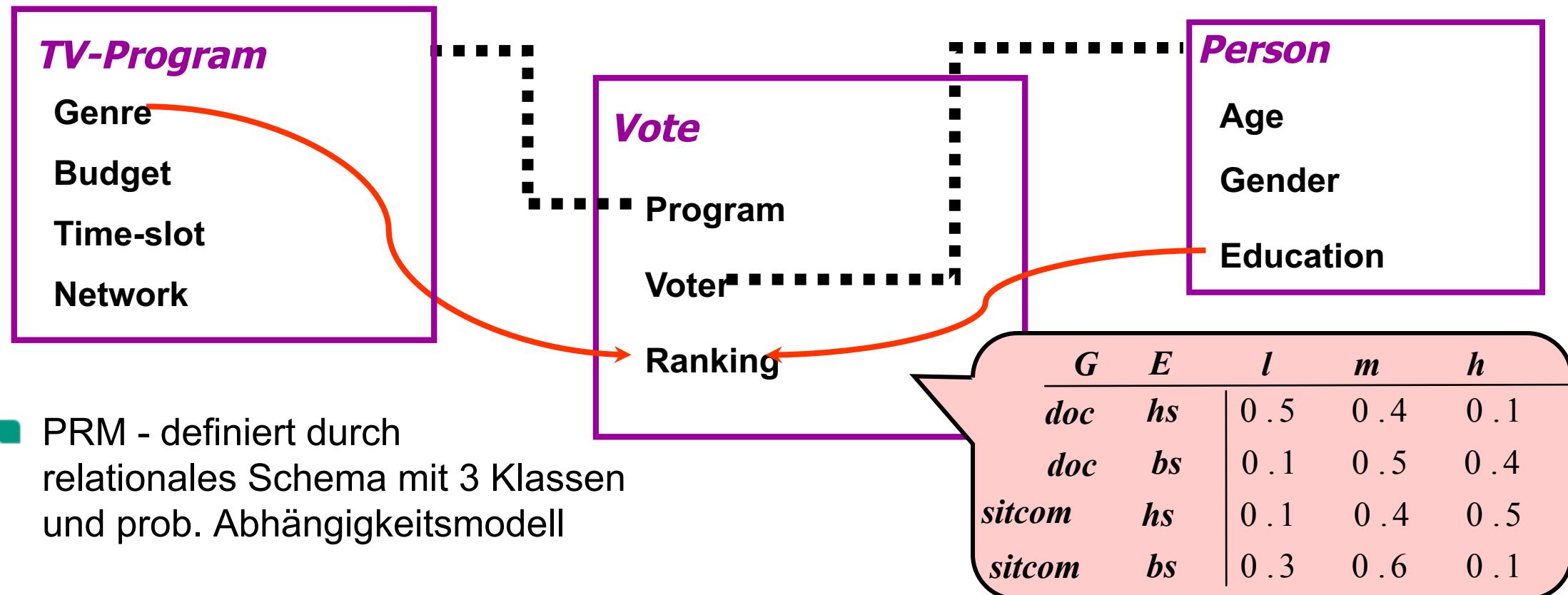
Problematik:

- In PRM verwenden alle Instanzen einer Klasse dieselben Abhängigkeiten
→ dies kann auch einschränkend sein

- Beispiel – siehe nächste Folien: Beliebtheit (Bewertung) von Fernsehprogrammen bei Personen
 - Problem: Gegebene Klasse Fernsehprogramm kann zunächst nicht weiter differenziert werden z.B. Dokumentarprogramm oder Sitcom

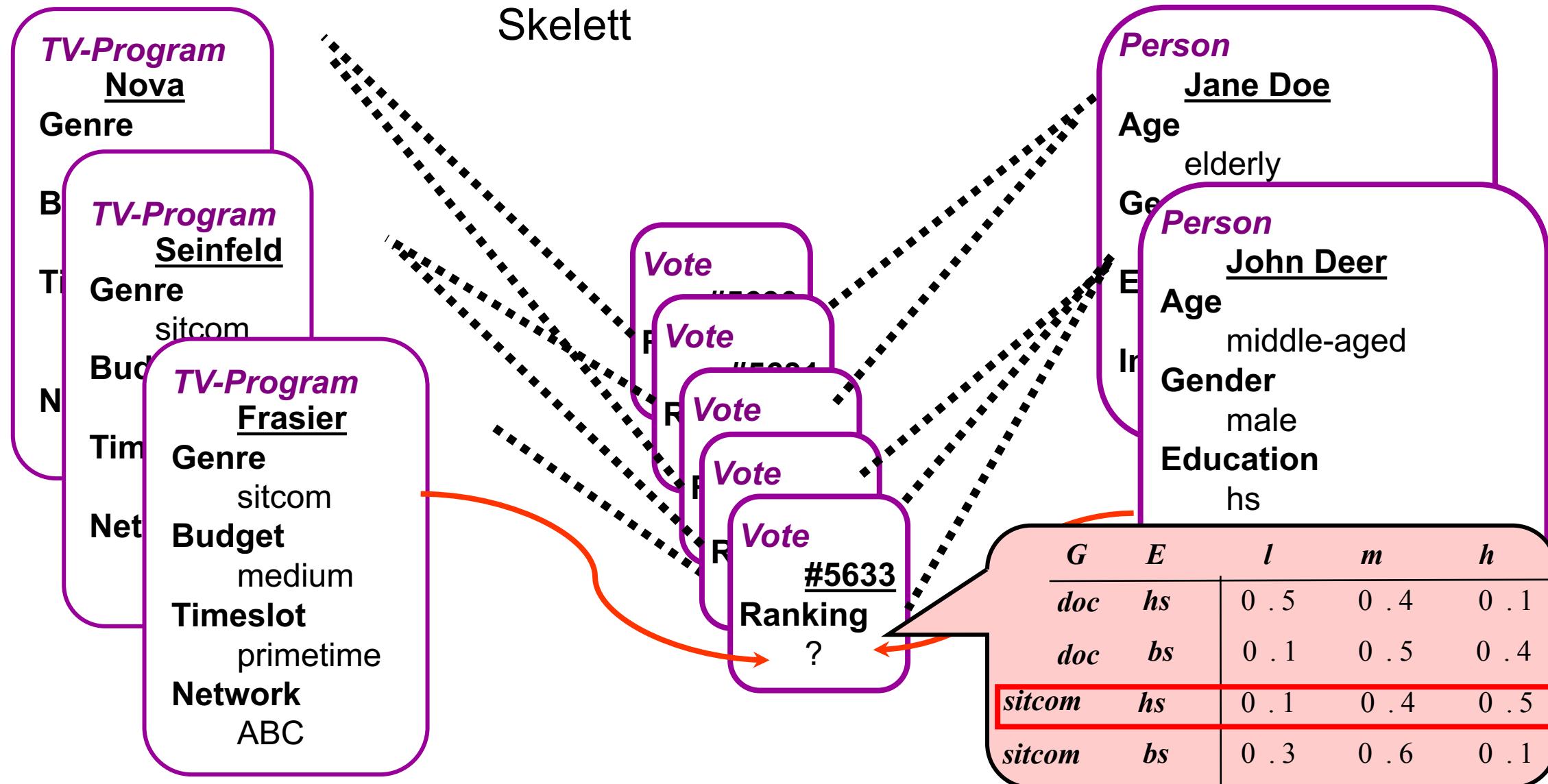
Motivation – Limitierung des PRM

- Beispiel: Bewerten von Fernsehprogrammen – einfaches Modell OK



- PRM - definiert durch relationales Schema mit 3 Klassen und prob. Abhängigkeitsmodell

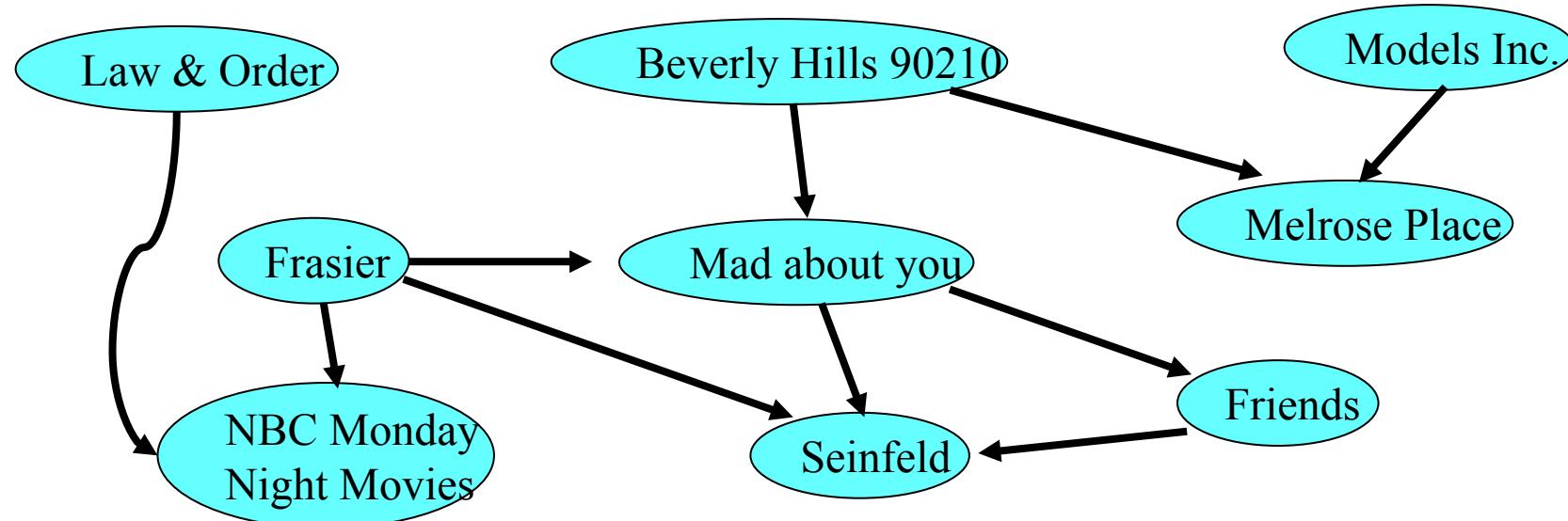
Motivation – Limitierung des PRM



Motivation – Limitierung

ABER komplexere Abhangigkeiten nicht moglich weil:

- PRM darf keine zyklischen Abhangigkeiten haben, d.h. z.B.:
 - Ranking fur ein Programm darf Ranking fur anderes Prog. nicht beeinflussen (sonst ware ein Attribut einer Klasse von sich abhangig)
 - Wurde aber in einem BN funktionieren



Motivation – Limitierung

ABER komplexere Abhängigkeiten nicht möglich weil:

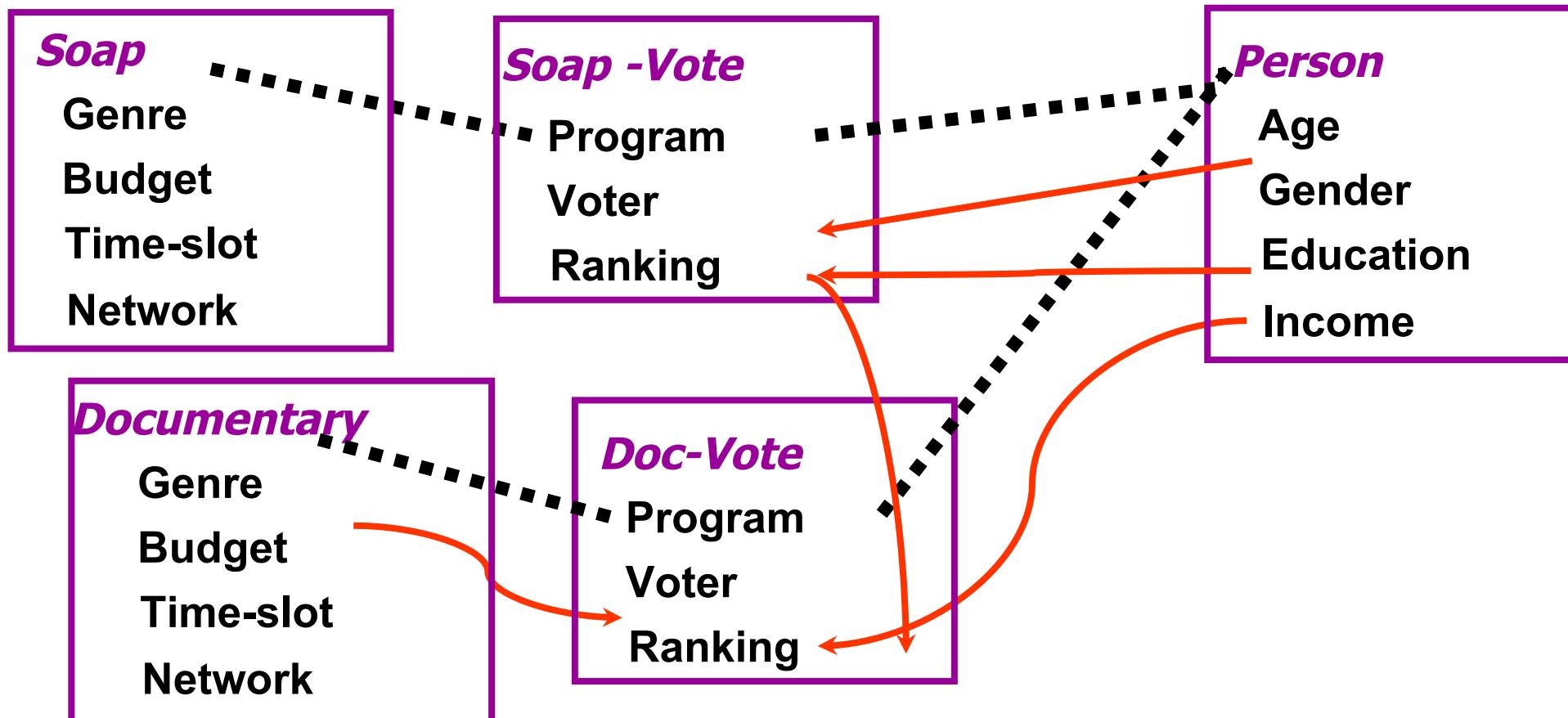
- PRM darf keine zyklischen Abhängigkeiten haben, d.h. z.B.:
 - Ranking für ein Programm darf Ranking für anderes Prog. nicht beeinflussen (sonst wäre ein Attribut einer Klasse von sich abhängig)

PRM – schlechter als BN ?

- Im Gegensatz zu PRM hat zwar in Bayes'schen Netzen jede Instanz eigene Abhängigkeiten ABER
 - eine Generalisierung ist nicht möglich
 - Z.B.: John liebt Soaps und daher auch Dokus ← geht nicht in BN
 - denn BN können keine Klassenabhängigkeiten modellieren

Gewünschtes (komplexes) Modell

- Klassen- und Instanzenabhängigkeit → Unterklassen

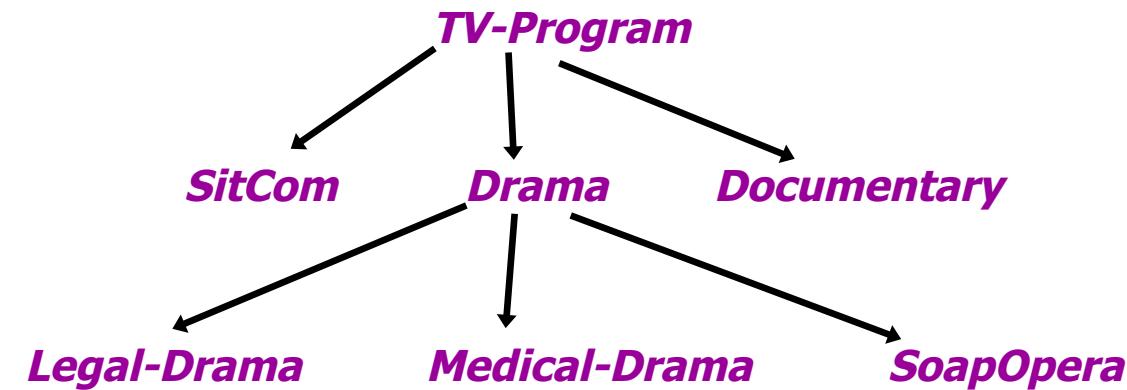
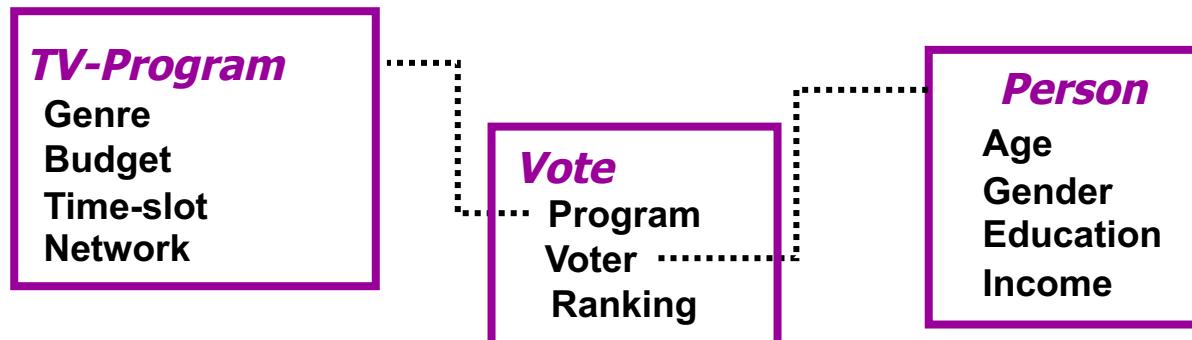


PRM mit Klassenhierarchien (PRM-CH)

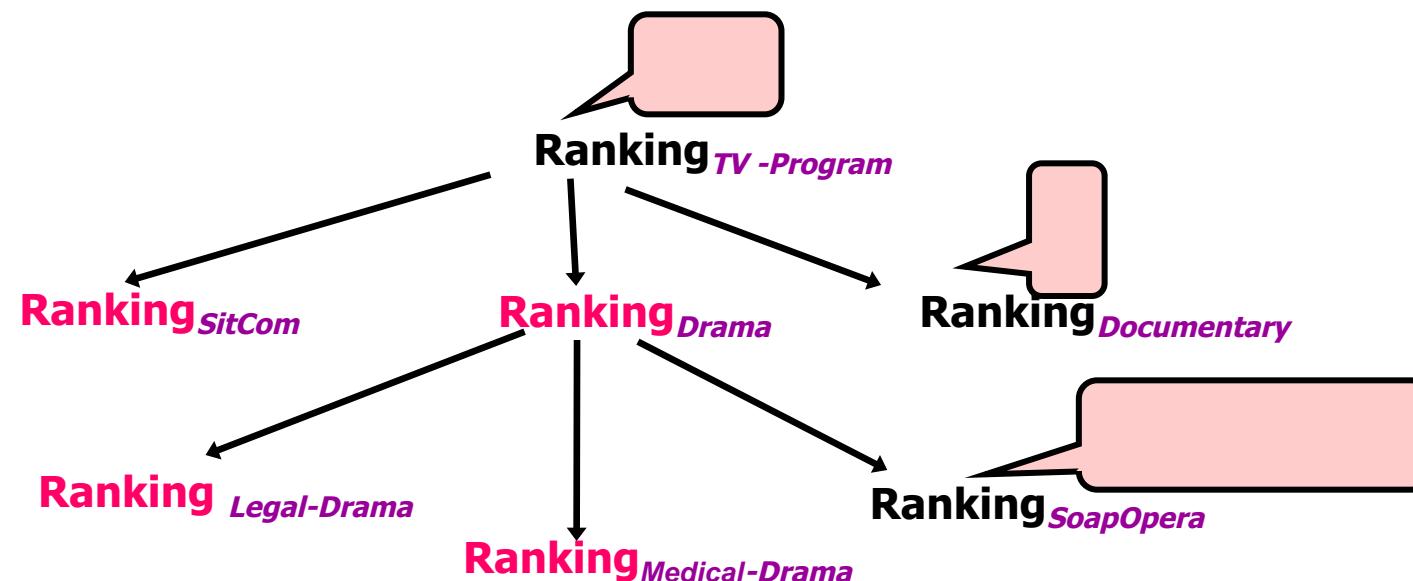
Idee:

- “Heterogene” Klassen in kohärentere Unterklassen aufteilen → Bilden von Klassenhierarchien
 - Vorteil: Beim Lernen probabilistisches Modell entlang der Klassenhierarchie weitergegeben
 - Spezialisierung und Vererbung von Wahrscheinlichkeiten
 - Neue Wahrscheinlichkeitstabellen (im diskreten Fall) durch neue Abhängigkeiten die azyklisch sind
- „Übergang“ vom Klassenmodell zum instanzbasierten Modell

PRM mit Klassen-Hierarchien



Relationales Schema

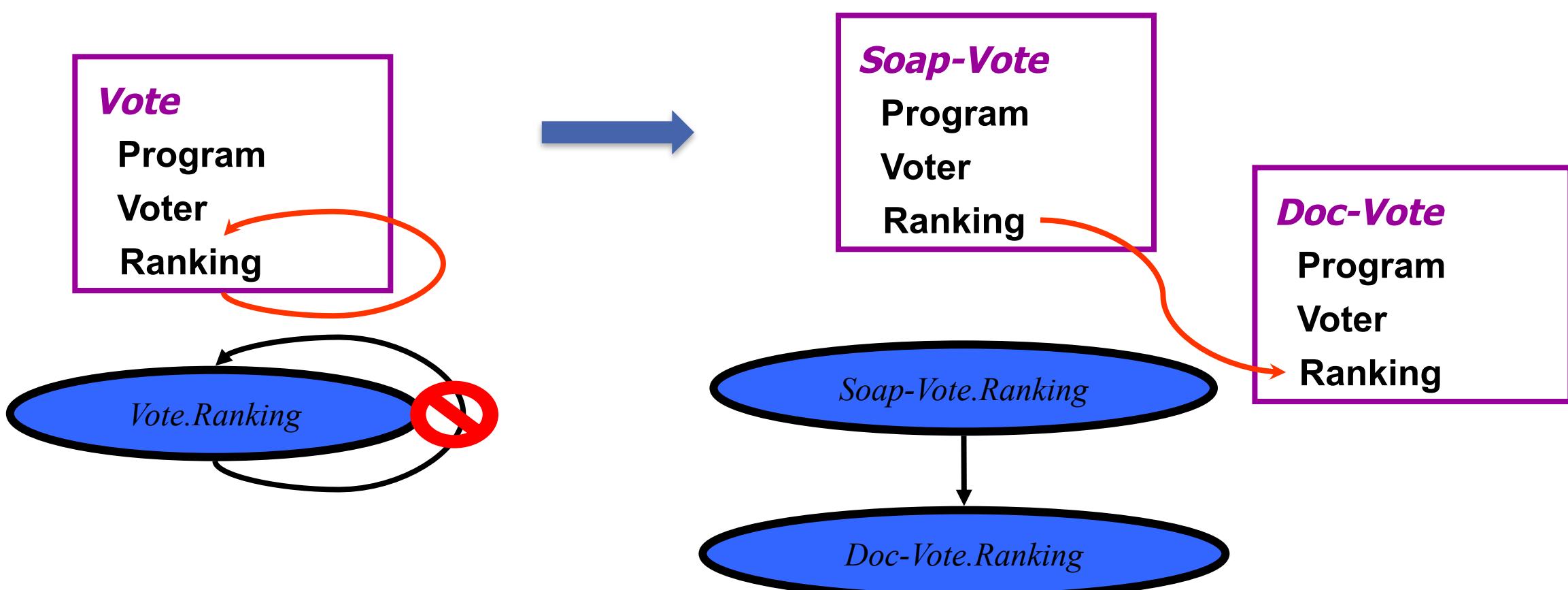


Klassenhierarchie

Abhängigkeitsmodell
unterschiedlich

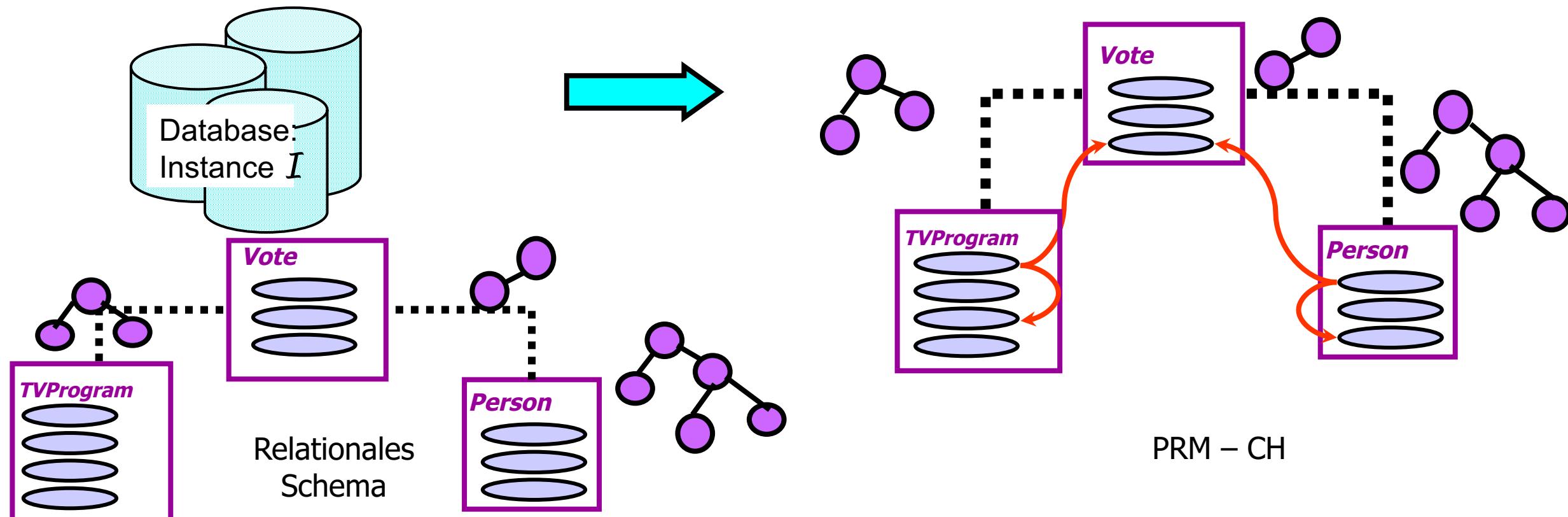
Azyklische Abhängigkeiten

- Durch Aufteilung nach Klassenhierarchien können azyklische Abhängigkeiten (z.B. in den Klassen) eingeführt werden



Lernen von PRM mit Klassenhierarchie

- Gegeben: Klassenhierarchie und Lerndaten
- Ziel: Lernen der Hierarchischen PRMs



Strukturbestimmung

- Idee:
 - Verwenden einer Bewertungsfunktion für PRM - Strukturen
 - Lokale Suche über gültige Strukturen

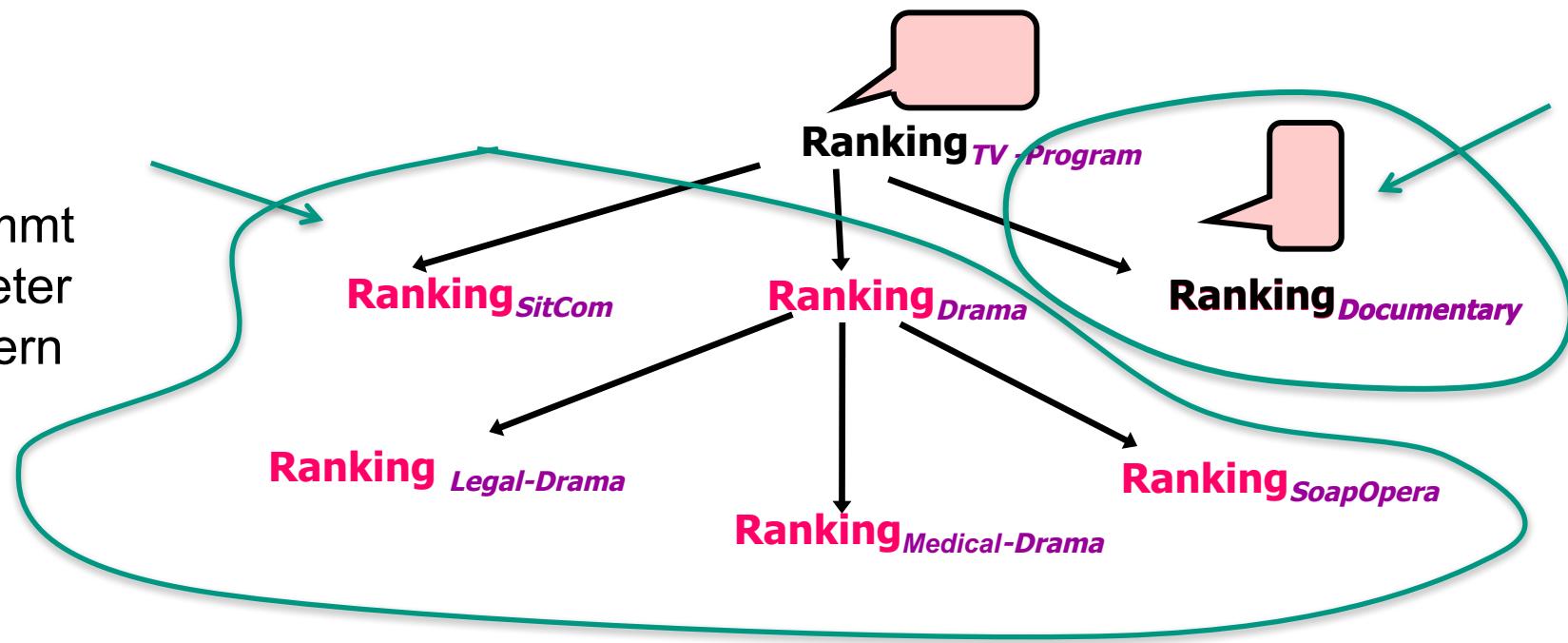
- Wesentliche Schritte:
 - Bestimmung gültiger Modelle
 - Unverändert !
 - Definition von Bewertungsfunktionen
 - Unverändert !
 - Klassenhierarchie
 - Gegeben
 - Lernen
 - Suche im Modellraum
 - Zusätzliche (Klassen-) Operatoren

Lernen mit Klassenhierarchie

- Klassenhierarchie gegeben
- Neue Operatoren
 - Spezialisierung und Erben der Modelle entlang der Klassenhierarchie

Erben

- Instanz übernimmt Parameter von Eltern



Spezialisierung

- Lernen der Parameter mit Daten dieser Klasse
- Zulassen neuer Relationen

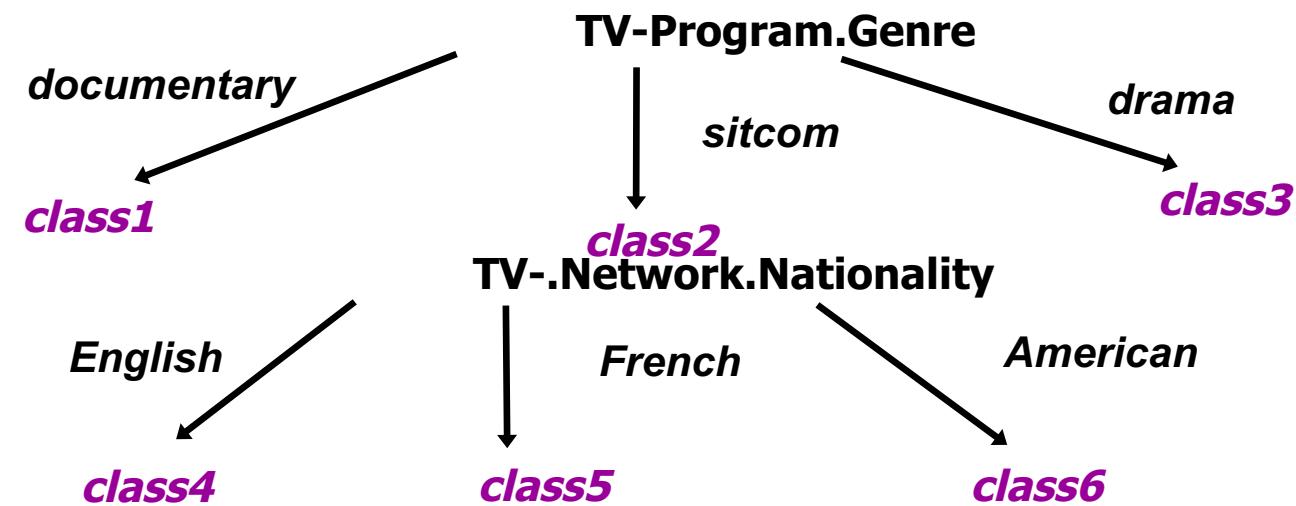
Lernen ohne Klassenhierarchien

- Klassenhierarchien finden → Baum für Klassen

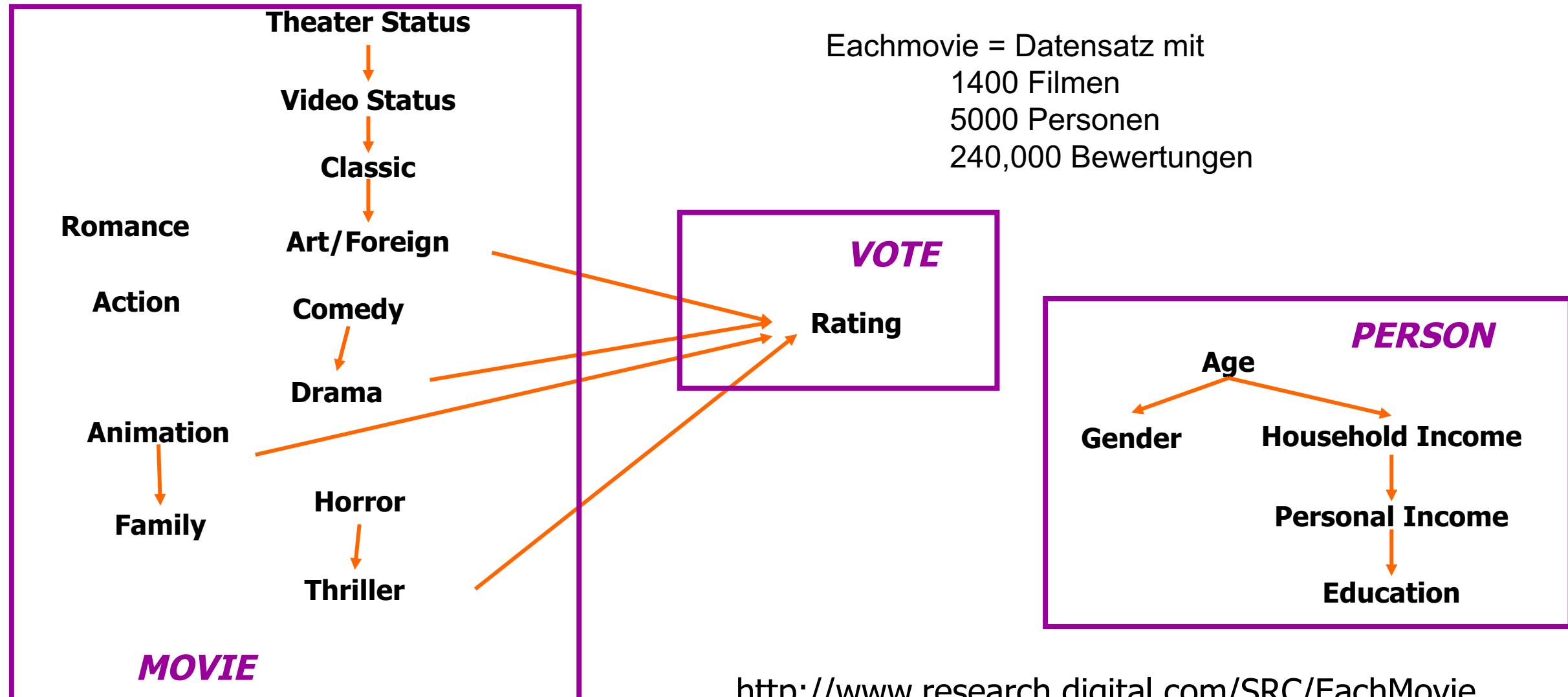
- Anhand der Attribute der eigenen Klasse oder
- Anhand der Attribute der anderen Klassen, z.B. Vote teilen anhand von Programm → Sitcom-Vote, Doc-Vote, ...
- Heuristisch aus Daten z.B. wenn ein Attribut viele Abhängigkeiten beeinflusst ist es ein potentieller Kandidat für einen guten Split

- Operatoren

- Spezialisierung und Vererbung verwenden den Klassen-Baum

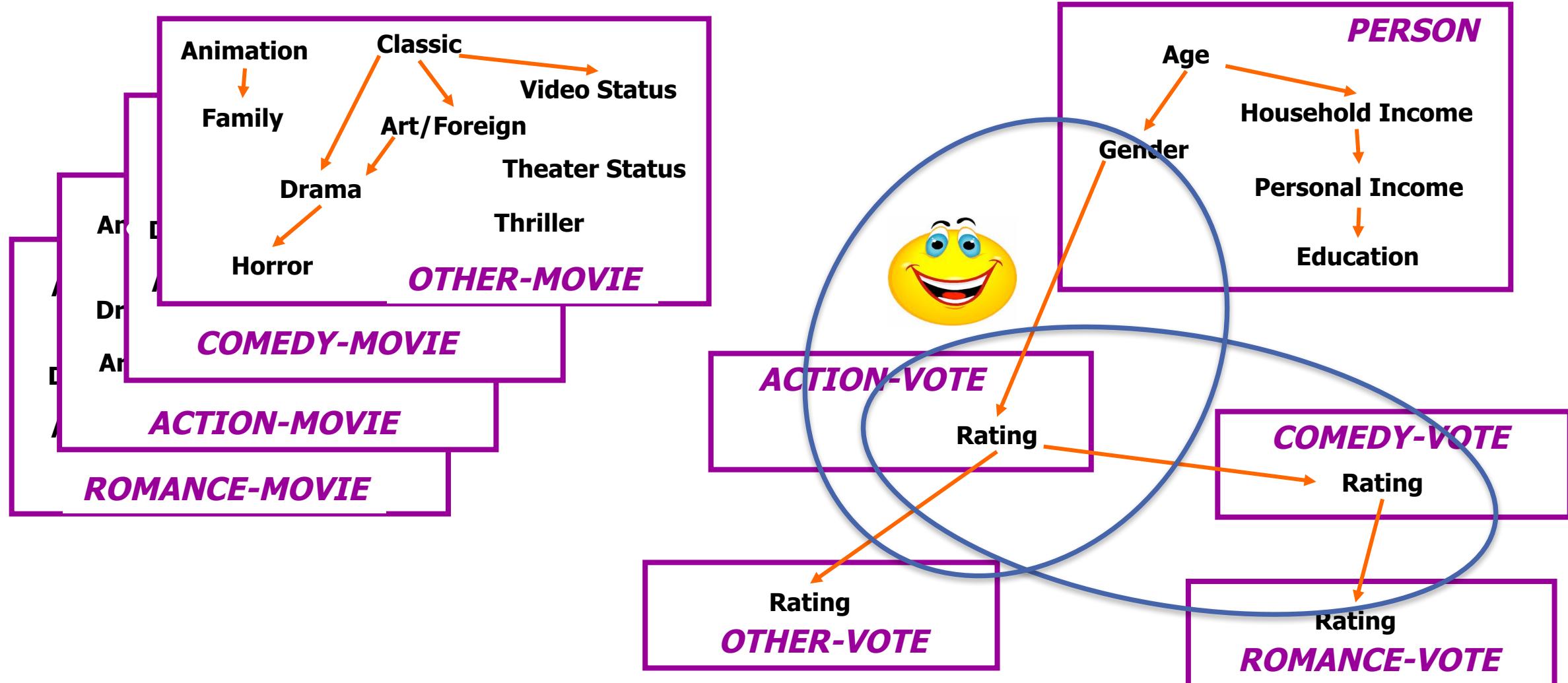


Beispiel: Anwendung auf EachMovie



<http://www.research.digital.com/SRC/EachMovie>

Beispiel: Einige Abhängigkeiten



Beispiel: (Qualitatives) Ergebnis

- 5 Testsätze:
 - 1000 Abstimmungen,
 - 100 Filme,
 - 115 Personen
- kleiner Datensatz
- PRM mit Klassenhierarchien deutlich besser, mit einer Güte von 99%, als einfaches PRM Model
- Dies ist ein „erstes“ Ergebnis → Allgemein: Validierung der Aussage auf realen Domänen nötig!

Anwendung – CoCar kognitives Automobil

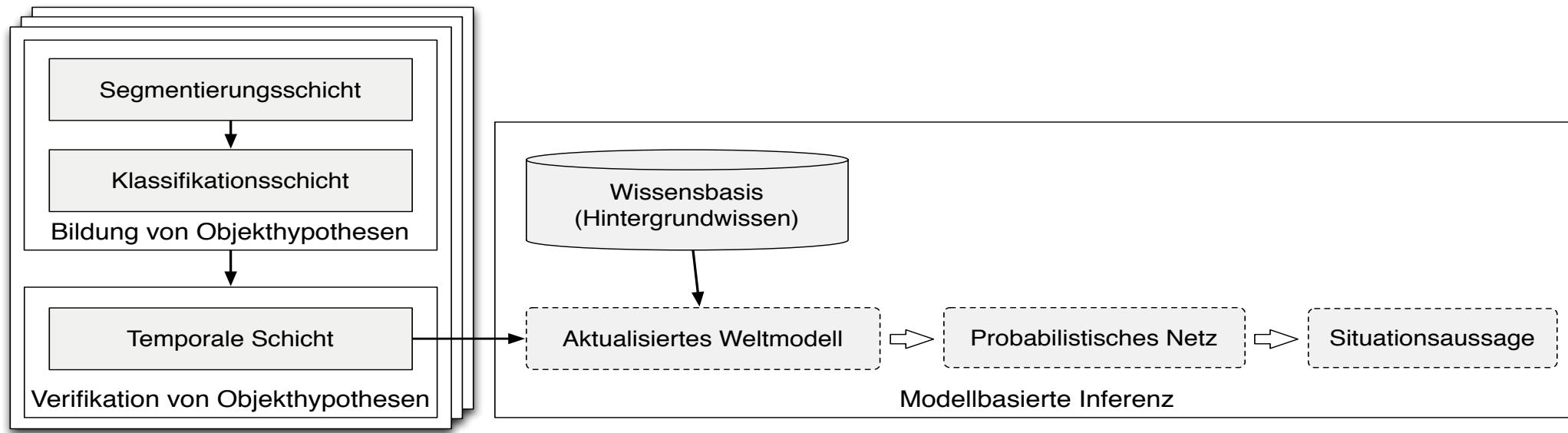


Situationsanalyse



Schamm et. al 2011

Konzept zur Situationsanalyse im Fahrzeug



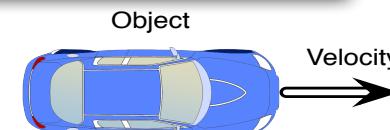
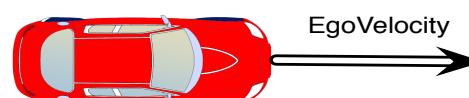
- Definition der Modellbasis, welche die relevanten Entitäten und Relationen enthält (spezifisch für Wahrnehmungskomponenten)
- Instanziierung des Weltmodells und automatische Konstruktion des graphischen Modells auf Grundlage der Sensorinformationen
- Inferenz der unbekannten Attributwerte zur Erfassung der Situationsaussage

Instanziierung des Weltmodells (Teilausschnitt)

- Modellierung Ego-Objekt (OPRML - Object-Oriented Probabilistic Relational Modelling Language)
- Relation zu weiteren Objekten
- Parameter der Modellinstanzen werden über die Zeit angepasst, Instanzen von veralteten Objekten werden entfernt

Class	EgoVehicle										
Superclass	Entity										
Id	ka_fz101										
Object	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>Object</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>Velocity</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	Object	Parents		Distribution		Imports	Velocity
Domain	EgoVehicle										
Range	Object										
Parents											
Distribution											
Imports	Velocity										
EgoVelocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, 0.25]</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, 0.25]		
Domain	EgoVehicle										
Range	zero, slow, medium, fast										
Parents											
Distribution	[0.25, 0.25, 0.25, 0.25]										
RelativeVelocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>depart, constant, slow approach, fast approach</td></tr> <tr> <td>Parents</td><td>Object.Velocity, EgoVelocity</td></tr> <tr> <td>Distribution</td><td>[0.0, 1.0, 0.0, ...]</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	depart, constant, slow approach, fast approach	Parents	Object.Velocity, EgoVelocity	Distribution	[0.0, 1.0, 0.0, ...]		
Domain	EgoVehicle										
Range	depart, constant, slow approach, fast approach										
Parents	Object.Velocity, EgoVelocity										
Distribution	[0.0, 1.0, 0.0, ...]										

Class	Object								
Superclass	Entity								
Id	object_42								
Velocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, 0.25]</td></tr> </tbody> </table>	Domain	Object	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, 0.25]
Domain	Object								
Range	zero, slow, medium, fast								
Parents									
Distribution	[0.25, 0.25, 0.25, 0.25]								

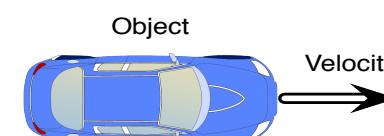
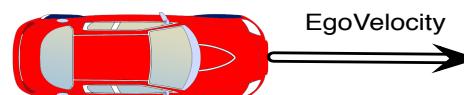
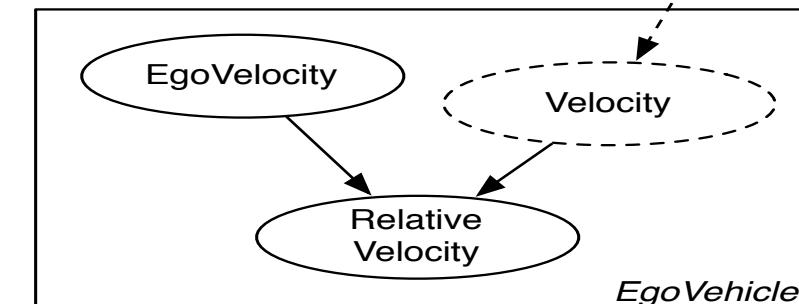
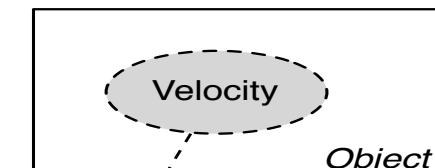


Konstruktion des graphischen Modells (Teilausschnitt)

- Die probabilistischen Attribute bilden ein graphisches Modell

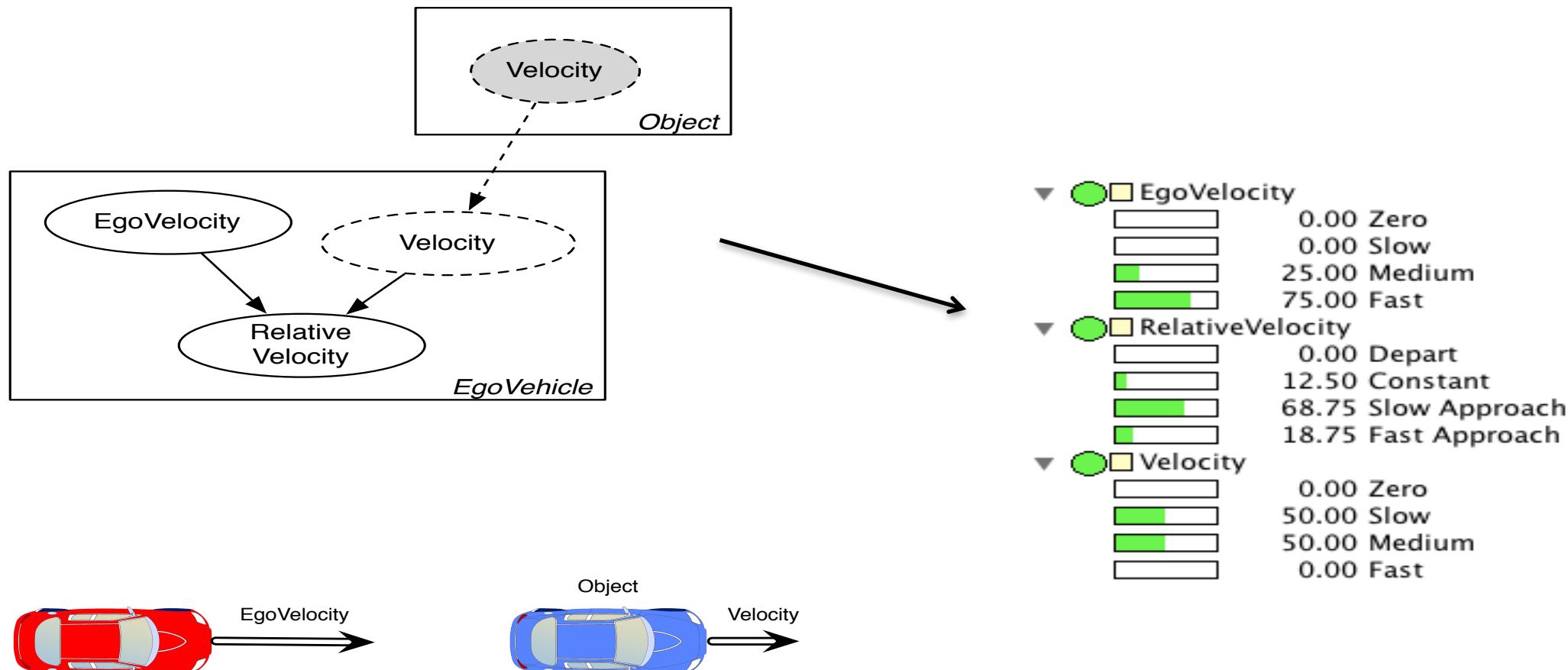
Class	EgoVehicle										
Superclass	Entity										
Id	ka_fz101										
Object	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>Object</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>Velocity</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	Object	Parents		Distribution		Imports	Velocity
Domain	EgoVehicle										
Range	Object										
Parents											
Distribution											
Imports	Velocity										
EgoVelocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, 0.25]</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, 0.25]		
Domain	EgoVehicle										
Range	zero, slow, medium, fast										
Parents											
Distribution	[0.25, 0.25, 0.25, 0.25]										
RelativeVelocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>depart, constant, slow approach, fast approach</td></tr> <tr> <td>Parents</td><td>Object.Velocity, EgoVelocity</td></tr> <tr> <td>Distribution</td><td>[0.0, 1.0, 0.0, ...]</td></tr> </tbody> </table>	Domain	EgoVehicle	Range	depart, constant, slow approach, fast approach	Parents	Object.Velocity, EgoVelocity	Distribution	[0.0, 1.0, 0.0, ...]		
Domain	EgoVehicle										
Range	depart, constant, slow approach, fast approach										
Parents	Object.Velocity, EgoVelocity										
Distribution	[0.0, 1.0, 0.0, ...]										

Class	Object								
Superclass	Entity								
Id	object_42								
Velocity	<table border="1"> <tbody> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, 0.25]</td></tr> </tbody> </table>	Domain	Object	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, 0.25]
Domain	Object								
Range	zero, slow, medium, fast								
Parents									
Distribution	[0.25, 0.25, 0.25, 0.25]								



Konstruktion des graphischen Modells (Teilausschnitt)

- Bestimmung der Marginalverteilung durch Belief Propagation

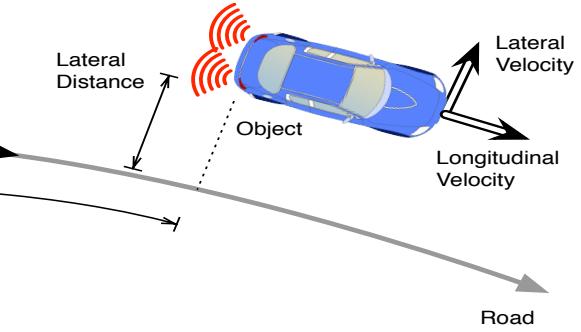


Anwendung – Bestimmung der Kollisionsrisikos

- Risiko = Eintrittswahrscheinlichkeit * Auswirkung

Class	Object								
Superclass	Entity								
Id	<Name>								
LongitudinalDistance	<table border="1"> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>close, medium, far, very far</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, ...]</td></tr> </table>	Domain	Object	Range	close, medium, far, very far	Parents		Distribution	[0.25, 0.25, 0.25, ...]
Domain	Object								
Range	close, medium, far, very far								
Parents									
Distribution	[0.25, 0.25, 0.25, ...]								
LongitudinalVelocity	<table border="1"> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, ...]</td></tr> </table>	Domain	Object	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, ...]
Domain	Object								
Range	zero, slow, medium, fast								
Parents									
Distribution	[0.25, 0.25, 0.25, ...]								
LateralDistance	<table border="1"> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>blockage, overlap, near, far</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, ...]</td></tr> </table>	Domain	Object	Range	blockage, overlap, near, far	Parents		Distribution	[0.25, 0.25, 0.25, ...]
Domain	Object								
Range	blockage, overlap, near, far								
Parents									
Distribution	[0.25, 0.25, 0.25, ...]								
LateralVelocity	<table border="1"> <tr> <td>Domain</td><td>Object</td></tr> <tr> <td>Range</td><td>approach, constant, depart</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, ...]</td></tr> </table>	Domain	Object	Range	approach, constant, depart	Parents		Distribution	[0.25, 0.25, 0.25, ...]
Domain	Object								
Range	approach, constant, depart								
Parents									
Distribution	[0.25, 0.25, 0.25, ...]								
↓									
Class	EgoVehicle								
Superclass	Entity								
Id	<Name>								
LongitudinalVelocity	<table border="1"> <tr> <td>Domain</td><td>EgoVehicle</td></tr> <tr> <td>Range</td><td>zero, slow, medium, fast</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td>[0.25, 0.25, 0.25, ...]</td></tr> </table>	Domain	EgoVehicle	Range	zero, slow, medium, fast	Parents		Distribution	[0.25, 0.25, 0.25, ...]
Domain	EgoVehicle								
Range	zero, slow, medium, fast								
Parents									
Distribution	[0.25, 0.25, 0.25, ...]								

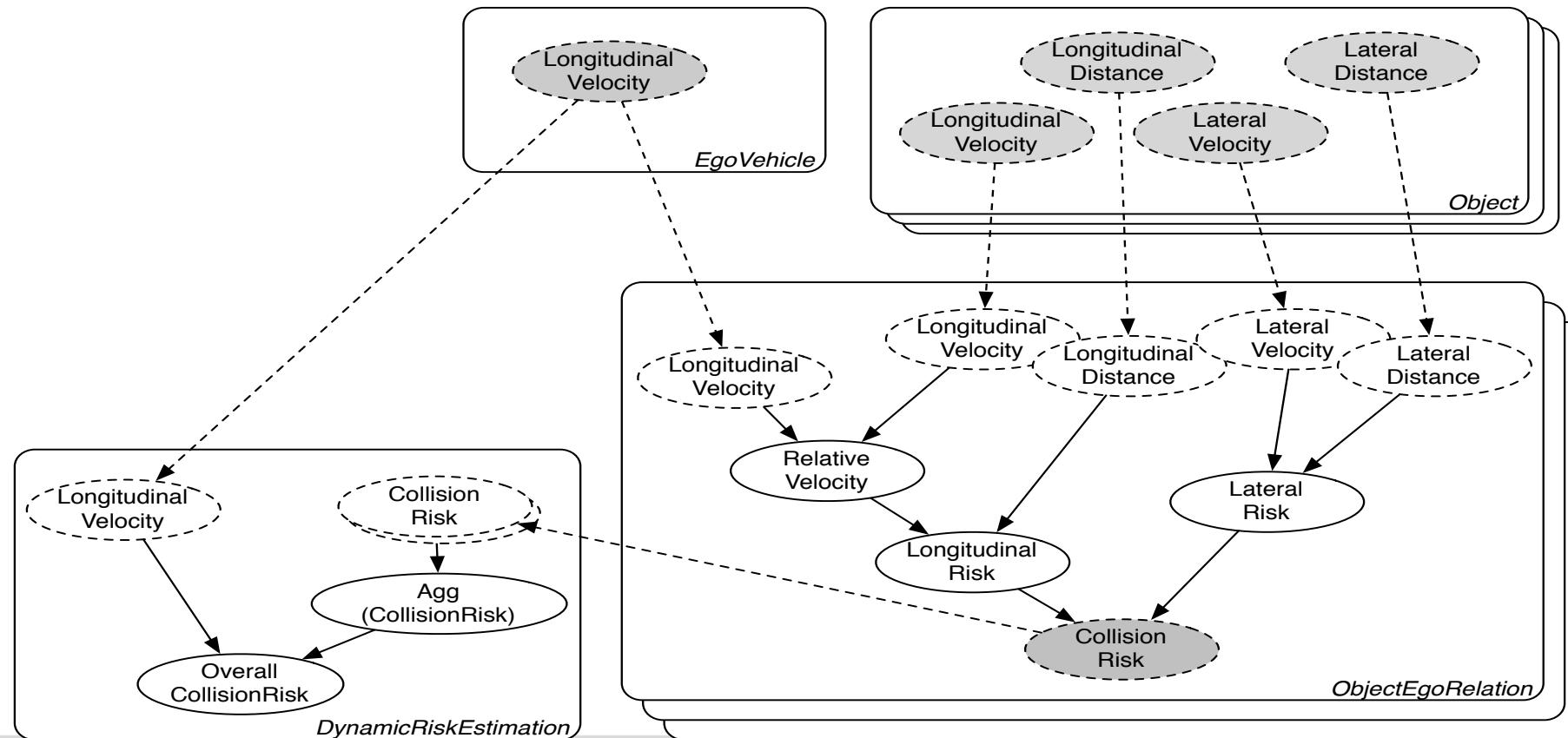
Class	ObjectEgoRelation										
Superclass	Relation										
Id	<Name>										
Object	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>Object</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>LongDistance, LongVelocity, LatDistance, LatVelocity</td></tr> </table>	Domain	ObjectEgoRelation	Range	Object	Parents		Distribution		Imports	LongDistance, LongVelocity, LatDistance, LatVelocity
Domain	ObjectEgoRelation										
Range	Object										
Parents											
Distribution											
Imports	LongDistance, LongVelocity, LatDistance, LatVelocity										
EgoVehicle	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>EgoVehicle</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>LongitudinalVelocity</td></tr> </table>	Domain	ObjectEgoRelation	Range	EgoVehicle	Parents		Distribution		Imports	LongitudinalVelocity
Domain	ObjectEgoRelation										
Range	EgoVehicle										
Parents											
Distribution											
Imports	LongitudinalVelocity										
RelativeVelocity	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>depart, constant, slow approach, fast approach</td></tr> <tr> <td>Parents</td><td>Object.LongitudinalVelocity, EgoVehicle.LongitudinalVelocity</td></tr> <tr> <td>Distribution</td><td>[0.0, 1.0, 0.0, ...]</td></tr> </table>	Domain	ObjectEgoRelation	Range	depart, constant, slow approach, fast approach	Parents	Object.LongitudinalVelocity, EgoVehicle.LongitudinalVelocity	Distribution	[0.0, 1.0, 0.0, ...]		
Domain	ObjectEgoRelation										
Range	depart, constant, slow approach, fast approach										
Parents	Object.LongitudinalVelocity, EgoVehicle.LongitudinalVelocity										
Distribution	[0.0, 1.0, 0.0, ...]										
LongitudinalRisk	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>zero, low, medium, high</td></tr> <tr> <td>Parents</td><td>RelativeVelocity, Object.LongitudinalDistance</td></tr> <tr> <td>Distribution</td><td>[1.0, 0.0, 0.0, ...]</td></tr> </table>	Domain	ObjectEgoRelation	Range	zero, low, medium, high	Parents	RelativeVelocity, Object.LongitudinalDistance	Distribution	[1.0, 0.0, 0.0, ...]		
Domain	ObjectEgoRelation										
Range	zero, low, medium, high										
Parents	RelativeVelocity, Object.LongitudinalDistance										
Distribution	[1.0, 0.0, 0.0, ...]										
LateralRisk	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>zero, low, medium, high</td></tr> <tr> <td>Parents</td><td>Object.LateralVelocity, Object.LateralDistance</td></tr> <tr> <td>Distribution</td><td>[0.05, 0.1, 0.3, ...]</td></tr> </table>	Domain	ObjectEgoRelation	Range	zero, low, medium, high	Parents	Object.LateralVelocity, Object.LateralDistance	Distribution	[0.05, 0.1, 0.3, ...]		
Domain	ObjectEgoRelation										
Range	zero, low, medium, high										
Parents	Object.LateralVelocity, Object.LateralDistance										
Distribution	[0.05, 0.1, 0.3, ...]										
CollisionRisk	<table border="1"> <tr> <td>Domain</td><td>ObjectEgoRelation</td></tr> <tr> <td>Range</td><td>zero, low, medium, high</td></tr> <tr> <td>Parents</td><td>LongitudinalRisk, LateralRisk</td></tr> <tr> <td>Distribution</td><td>[0.05, 0.1, 0.3, ...]</td></tr> </table>	Domain	ObjectEgoRelation	Range	zero, low, medium, high	Parents	LongitudinalRisk, LateralRisk	Distribution	[0.05, 0.1, 0.3, ...]		
Domain	ObjectEgoRelation										
Range	zero, low, medium, high										
Parents	LongitudinalRisk, LateralRisk										
Distribution	[0.05, 0.1, 0.3, ...]										



Class	DynamicRiskEstimation										
Superclass	RiskClass										
Id	<Name>										
ObjectEgoRelation	<table border="1"> <tr> <td>Domain</td><td>DynamicRiskEstimation</td></tr> <tr> <td>Range</td><td>ObjectEgoRelation</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>CollisionRisk</td></tr> </table>	Domain	DynamicRiskEstimation	Range	ObjectEgoRelation	Parents		Distribution		Imports	CollisionRisk
Domain	DynamicRiskEstimation										
Range	ObjectEgoRelation										
Parents											
Distribution											
Imports	CollisionRisk										
EgoVehicle	<table border="1"> <tr> <td>Domain</td><td>DynamicRiskEstimation</td></tr> <tr> <td>Range</td><td>EgoVehicle</td></tr> <tr> <td>Parents</td><td></td></tr> <tr> <td>Distribution</td><td></td></tr> <tr> <td>Imports</td><td>LongitudinalVelocity</td></tr> </table>	Domain	DynamicRiskEstimation	Range	EgoVehicle	Parents		Distribution		Imports	LongitudinalVelocity
Domain	DynamicRiskEstimation										
Range	EgoVehicle										
Parents											
Distribution											
Imports	LongitudinalVelocity										
Agg(CollisionRisk)	<table border="1"> <tr> <td>Domain</td><td>DynamicRiskEstimation</td></tr> <tr> <td>Range</td><td>zero, low, medium, high</td></tr> <tr> <td>Parents</td><td>ObjectEgoRelation.CollisionRisk</td></tr> <tr> <td>Distribution</td><td>[1.0, 0.0, 0.0, ...]</td></tr> </table>	Domain	DynamicRiskEstimation	Range	zero, low, medium, high	Parents	ObjectEgoRelation.CollisionRisk	Distribution	[1.0, 0.0, 0.0, ...]		
Domain	DynamicRiskEstimation										
Range	zero, low, medium, high										
Parents	ObjectEgoRelation.CollisionRisk										
Distribution	[1.0, 0.0, 0.0, ...]										
OverallCollisionRisk	<table border="1"> <tr> <td>Domain</td><td>DynamicRiskEstimation</td></tr> <tr> <td>Range</td><td>zero, low, medium, high</td></tr> <tr> <td>Parents</td><td>Agg(CollisionRisk), EgoVehicle.LongitudinalVelocity</td></tr> <tr> <td>Distribution</td><td>[1.0, 0.0, 0.0, ...]</td></tr> </table>	Domain	DynamicRiskEstimation	Range	zero, low, medium, high	Parents	Agg(CollisionRisk), EgoVehicle.LongitudinalVelocity	Distribution	[1.0, 0.0, 0.0, ...]		
Domain	DynamicRiskEstimation										
Range	zero, low, medium, high										
Parents	Agg(CollisionRisk), EgoVehicle.LongitudinalVelocity										
Distribution	[1.0, 0.0, 0.0, ...]										

Anwendung – Bestimmung der Kollisionsrisikos

- Aggregation mehrerer Attribute zu einem Attribut
 - Sum, Max, Min, Avg, Mode, Count



Anwendung – Bestimmung der Kollisionsrisikos



Anwendung – Blockadewahrscheinlichkeit durch Fußgänger

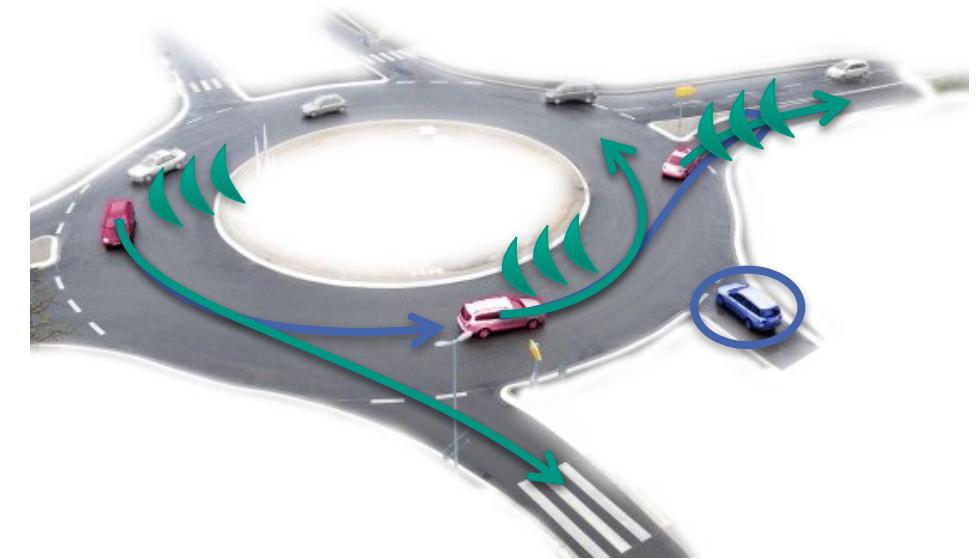


Schamm, Kuhnt et. al 2012

Situationsinterpretation, Verhaltensentscheidung

Herausforderungen:

- Implizite Kooperation
- Komplexität steigt stark mit Anzahl Objekte
- Kontinuierlicher Zustandsraum
- Sehr hohe Anzahl an Situationen und möglichen Verläufen
- Situationen unvollständig beobachtbar und nur mit Unsicherheit prädizierbar

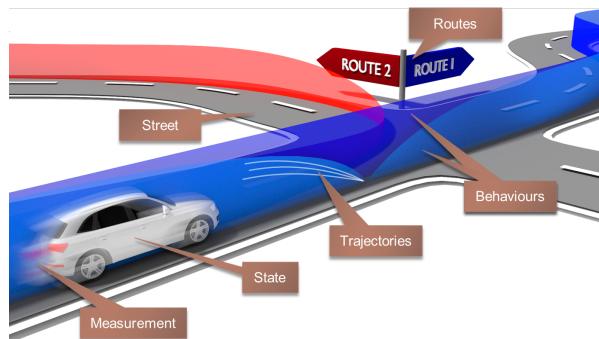


Konzeptuelle Ansätze

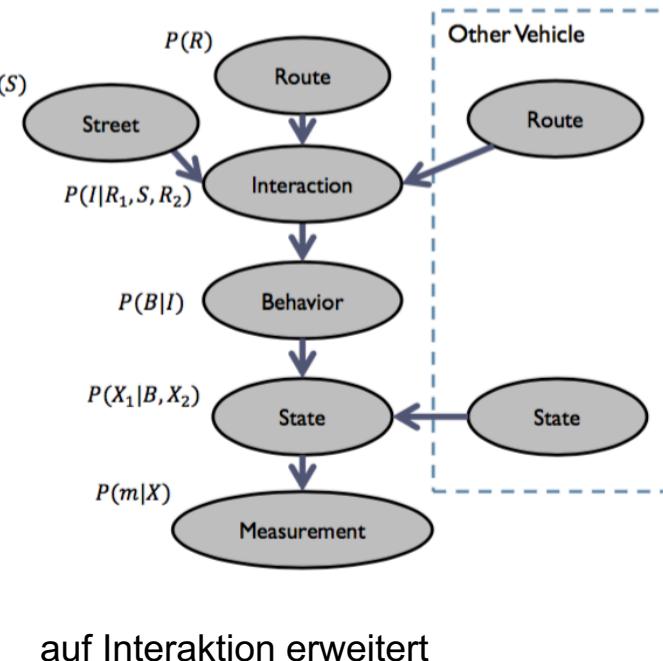
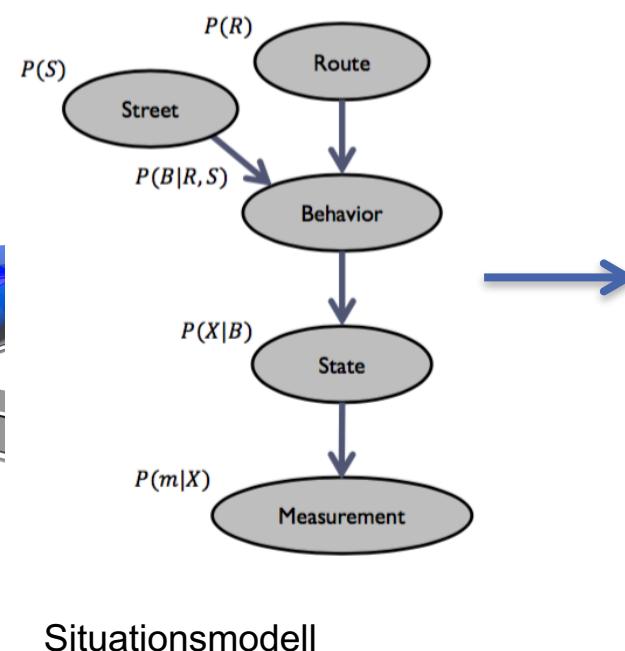
- Manöverplanung unter Berücksichtigung der Kooperation
- Explizite Schätzung der Kooperation
- Beherrschung der Systemkomplexität

Zugrundeliegendes Konzept

- Einsatz von Objektorientierten Probabilistisch Relationalen Modellen (OPRM)
 - Vorteile der formalen Logik verknüpft mit wahrscheinlichkeitsbasierter Inferenz und Objektorientierung
- Anwendung des Konzepts am Beispiel der Schätzung von Interaktionen

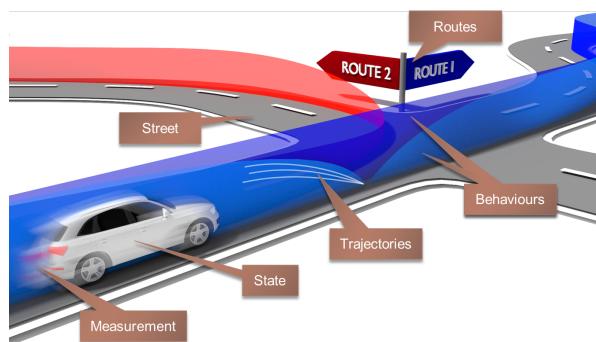


[Kuhnt, Fusion15]

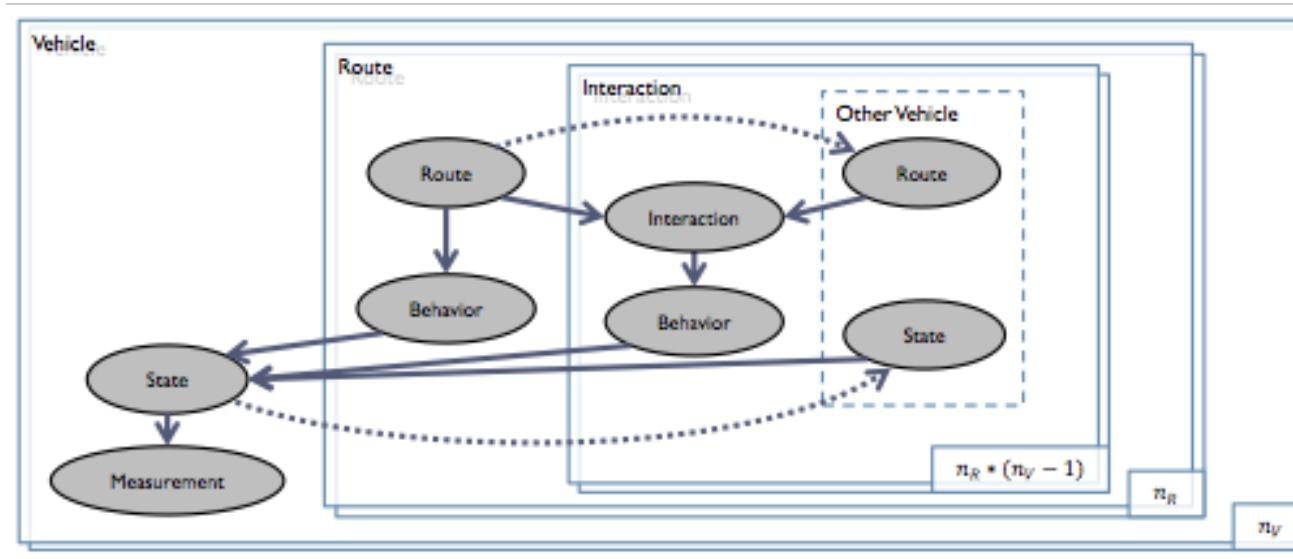


Zugrundeliegendes Konzept

- Einsatz von Objektorientierten Probabilistisch Relationalen Modellen (OPRM)
 - Vorteile der formalen Logik verknüpft mit wahrscheinlichkeitsbasierter Inferenz und Objektorientierung
- Anwendung des Konzepts am Beispiel der Schätzung von Interaktionen



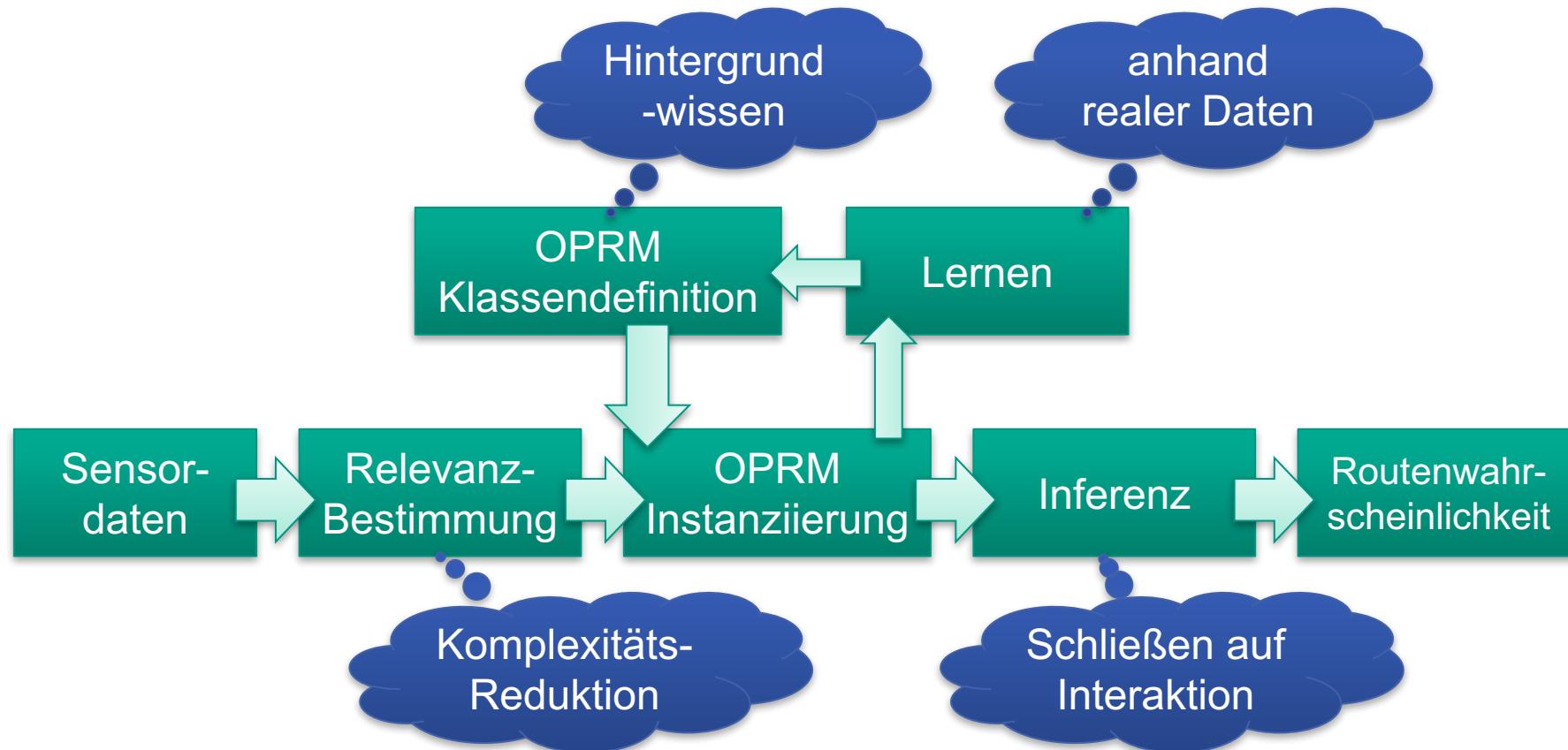
[Kuhnt, Fusion15]



Situationsmodell

auf Interaktion erweitert

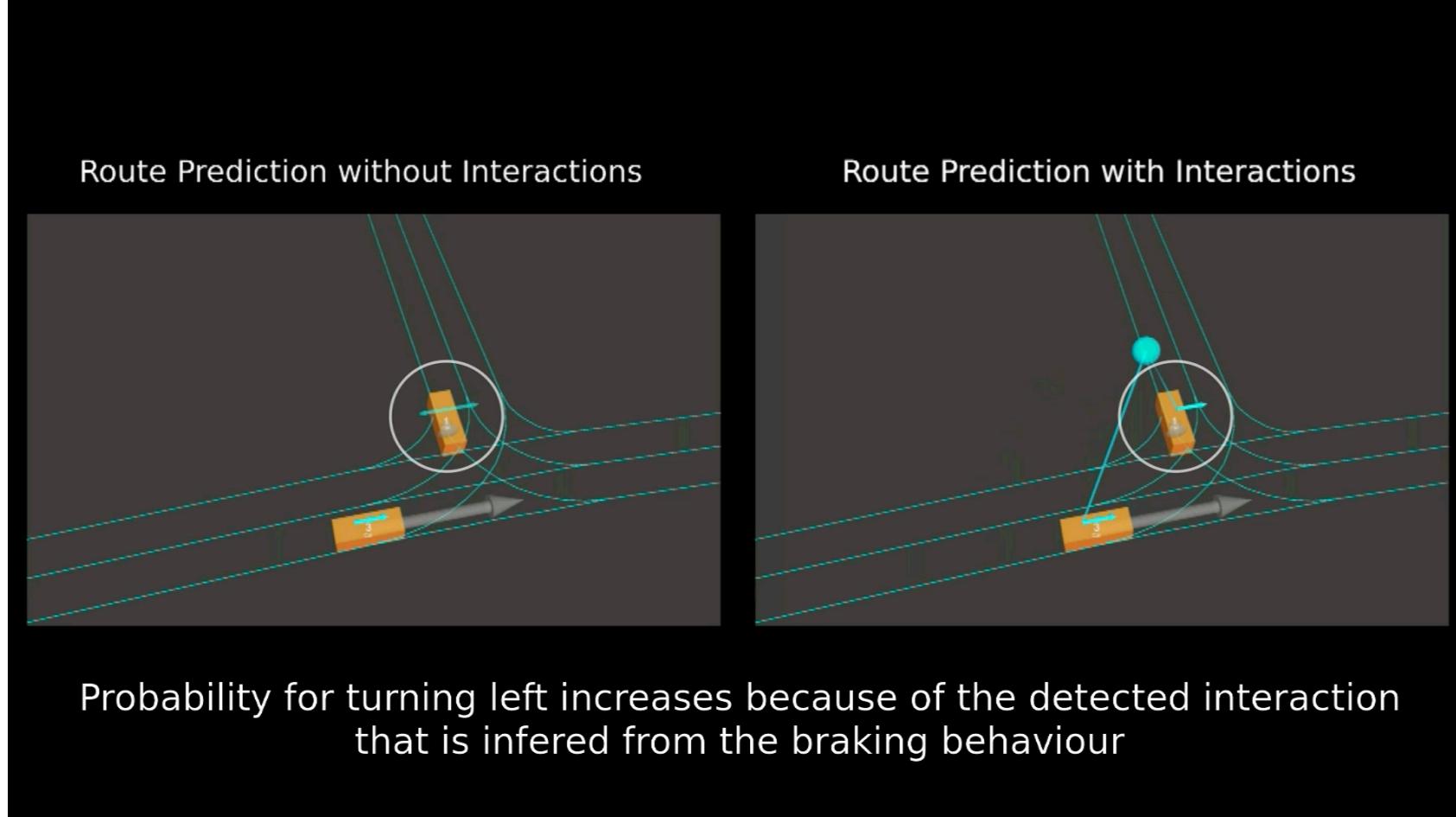
Ablauf der Schätzung von Interaktionen



[Kuhnt,16]

Weitere Beispiele – Interaktion und Prädiktion

- Geschätzte Interaktion (Kugel) & Fahrtrichtung (Pfeil)
(Limitierung – keine temporale Filterung)



Zusammenfassung

- PRMs erweitern Bayes'sche Netze um relationale Strukturen, **Klassen**, ...
- PRMs erweitern relationale Schemata um probabilistische Elemente
- Mit den Erweiterungen (strukturelle Unsicherheit und Klassenhierarchie) kann man Zusammenhänge sehr weitreichend und flexibel modellieren
- Inferenz – real nur approximativ möglich (siehe BN), kann aber effizienter gestaltet werden
- Ein PRM kann (effizient) aus einer Lerndatenmenge gelernt werden mit Klassenhierarchien auch in Abhängigkeit der Daten pro Klasse
- Mit der OPRML – existiert eine Erweiterung zur formalen Beschreibung
- Validierung und Übertragung auf reale Anwendungen nötig
 - Z.B. Situationsinterpretation für autonome und teil-autonome Systeme
- Verbesserung des Lernens (auf verschiedene Ebene) nötig

Literatur: Objektorientierte Probabilistisch Relationale Modelle

L. Getoor, and B. Taskar, editors: „Introduction to Statistical Relational Learning.“
The MIT Press, November 2007.

C. Howard, M. Stumptner: Automated compilation of Object-Oriented Probabilistic Relational Models,
International Journal of Approximate Reasoning, 2009

T. Schamm: Modellbasierter Ansatz zur probabilistischen Interpretation von Fahrsituationen, 2014,
Dissertation

T. Schamm et. al: A Model-based Approach to Probabilistic Situation Assessment for Driver Assistance Systems, 2011

Florian Kuhnt: Probabilistische Kollisionsprädiktion für Segway-Transporter, 2012

Florian Kuhnt et. al: Understanding interactions between traffic participants based on learned behaviors , 2016

ANHANG – DYNAMISCHE BAYESSCHE NETZE