Introduction to Data and Knowledge Engineering

Beispielklausur

Matrikelnummer

Nachname

Vorname

Fachbereich

Semester

Aufbaustudium ☐

Erreichte Punkte								
1	2	3	4	5	6	7	8	$\sum$
								/100

Viel Erfolg!

1 Datenbanken

10 1. Erklären Sie kurz und präzise die Begriffe

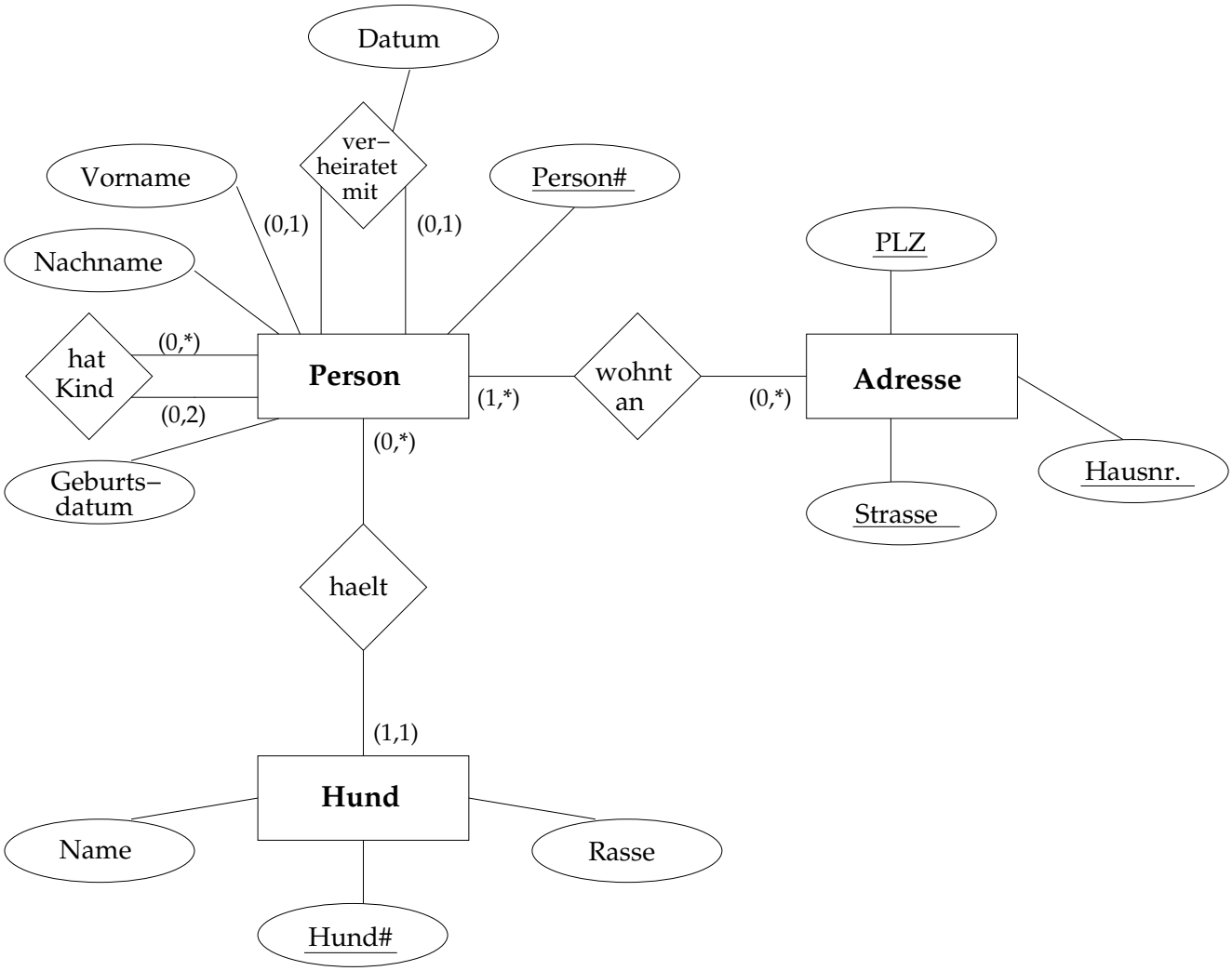
   1. Superschlüssel
   2. Schlüsselkandidat
   3. Primärschlüssel
   4. Fremdschlüssel
   5. partieller Schlüssel (oder Teilschlüssel).

Lösung:

1. Menge von Attributen, die ein Tupel identifiziert.
2. Minimaler Superschlüssel.
3. Ein beim Entwurf ausgewählter Schlüsselkandidat.
4. Eine Referenz auf einen gültigen Primärschlüssel.
5. Identifiziert ein Weak-Entity innerhalb seines zugehörigen Strong-Entitys.

12 2. Eine Stadtverwaltung mächte ihre Bärger in einer Datenbank speichern. Dazu sollen alle Personen mit Vor- und Nachnamen sowie dem Geburtsdatum erfasst werden. Falls 2 Personen miteinander verheiratet sind, soll dies inklusive dem Datum der Eheschliessung erfasst werden. Auäerdem wird gespeichert, welche Personen in einer Eltern-Kind-Beziehung zueinander stehen. In der Datenbank wird jede Adresse der Stadt erfasst, auch wenn sie nicht bewohnt ist. Jede Person muss unter mindestens einer Adresse gemeldet sein und an einer Adresse känen auch mehrere Personen wohnen. Ausserdem wird die Hundehaltung miterfasst. Jeder Hund hat eine Person als Halter und wird mit seinem Rufnamen und seiner Rasse gespeichert.

Identifizieren Sie die Entities und ihre Attribute und erstellen Sie einen ERM-Entwurf des Systems inklusive Komplexitäten. Ergänzen Sie, nur falls notwendig, Attribute zur eindeutigen Identifikation der Entities.



3. Gegeben seien die Relationen Klausur(KID, Vorlesung, Jahr, Maximalpunktzahl, Minimalpunktzahl) und Pruefungsleistung(PID, Student, Klausur, Punktzahl). In einer Pruefungsleistung wird erfasst, welche Punktzahl ein Student in einer bestimmten Klausur erzielt hat. Erzielt ein Student weniger als die Minimalpunktzahl, ist er durchgefallen. Das Attribut *Klausur* in der Relation Pruefungsleistung ist ein Fremdschlässel auf *KID* in Klausur.

6 (a) Formulieren Sie folgende Anfrage im Relationentupelkalkäl, in Relationaler Algebra und in SQL: Welche Maximalpunktzahl hätte man in der Präfungsleistung mit der PID 4711 erzielen kännen?

$\{t^{(1)} | (\exists p)(\exists k)(Pruefungsleistung(p) \wedge Klausur(k) \wedge p[3] = k[1] \wedge p[1] = 4711 \wedge t[1] = k[4])\}$

$\pi_{M}aximalpunktzahl(\sigma_{PID=4711}(Klausur \bowtie_{KID=Klausur} Pruefungsleistung))$

SELECT Maximalpunktzahl
FROM Klausur, Pruefungsleistung
WHERE Klausur = KID
AND PID = 4711;

Formulieren Sie folgende Anfragen in SQL

3 (b) Wie viele Studenten sind im Jahr 2007 in der Klausur zur Vorlesung ”DKE” durchgefallen?

3 (c) Erstellen Sie eine Liste mit der durchschnittlich erzielten Punktzahl aller Präfungsleistungen eines Jahres der Vorlesung ”DKE”. Die Liste soll das Jahr und die Durchschnittspunktzahl dieses Jahres enthalten-

5 (d) Erniedrigen Sie bei allen Präfungsleistungen zur Vorlesung ”DKE” im Jahr 2007, die genau die Mindestpunktzahl erreicht haben, die Punktzahl um 1.

Lösung:

b)

```
SELECT COUNT(*)
FROM Klausur, Pruefungsleistung
WHERE Klausur = KID
AND Vorlesung = 'DKE'
AND Jahr = 2007
AND Minimalpunktzahl > Punktzahl;
```

c)

```
SELECT Jahr, AVG(Punktzahl)
FROM Klausur, Pruefungsleistung
WHERE Klausur = KID
AND Vorlesung = 'DKE'
GROUP BY (Jahr);
```

d)

```
UPDATE Pruefungsleistung
SET Punktzahl = Punktzahl − 1
WHERE PID IN (
        SELECT PID
        FROM Klausur, Pruefungsleistung
        WHERE Klausur = KID
        AND Punktzahl = Mindestpunktzahl
        AND Vorlesung = 'DKE'
        AND Jahr = 2007
);
```

11 4. Sei $R = abcdefg$ eine Relation und $F = \{a \to c, b \to de, c \to f, f \to g, fg \to a, e \to g, a \to f\}$ eine Menge FDs über dieser Relation. Bestimmen Sie die Schlässelkandidaten, die hächste vorliegende Normalform (bis 3NF) und wenden Sie das vereinfachte Syntheseverfahren mit Dummy-FD an. überpräfen Sie anschlieäend, ob das Ergebnis die 3NF erfällt. Falls nicht, geben sie an welche NF verletzt ist, den Grund der Verletzung und welche Änderung nötig ist um die 3NF zu erreichen.

Lösung:

Schlässelkandidaten: $K = \{\{a, b\}, \{b, c\}, \{b, f\}\}$

Hächste erfällte Normalform: 1NF (z.B. ist d partiell von ab abhängig)

Vereinfachtes Syntheseverfahren:

Dummy-FD hinzufägen:
$abcdefg \to \delta$
$a \to c$
$a \to f$
$b \to de$
$c \to f$
$f \to g$
$fg \to a$
$e \to g$

Entfernen überflüssiger Attribute:
In $fg \to a$ ist g überflüssig.
In $abcdefg \to \delta$ sind entweder $cdefg$, $adefg$ oder $acdeg$ überflüssig.

Entfernen überflüssiger FDs:
$a \to f$ ist überflüssig.

Bilden der Äquivalenzklassen:
Die linken Seiten a, c und f sind äquivalent.
$ab \to \delta$
$a \to c$ $\quad c \to f$ $\quad f \to g$ $\quad f \to a$
$b \to de$
$e \to g$

Bilden der Relationen:
$R1 = (ab, \{\{a, b\}\})$
$R2 = (acfg, \{\{a\}, \{c\}, \{f\}\})$
$R3 = (bde, \{\{b\}\})$
$R4 = (eg, \{\{e\}\})$

3NF erfällt.

2 Knowledge Engineering

5. **Datalog**

Gegeben seien folgende Relationen:

```
zug(Zugnummer,Zugtyp)
haelt_in(Zugnummer, Ort, Bahnsteig, Ankunftszeit, Abfahrtszeit)
```

Hinweise:

- In Ihren Antworten können Sie alle Relationen durch deren Anfangsbuchstaben abkürzen.
- Die Ankunfts- und Abfahrtszeiten sind als Integer im Format *StundeStundeMinuteMinute* gespeichert, d.h. z.B. 9:07 Uhr ist repraesentiert als Zahl 907.
- Wir ignorieren das Problem, wie Anfangs- und Endbahnhöfe dargestellt werden, d.h. Sie können davon ausgehen, daß zu jedem Bahnhof eine Ankunfts- und Abfahrtszeit existiert.

3 (a) Definieren Sie in Datalog eine Relation `direktverbindung(A,B)`, die alle Orte A und B umfaßt, zwischen denen eine direkte Verbindung (d.h. ohne Umsteigen) existiert. Ob der Zug hierbei von A nach B oder von B nach A fährt ist egal.

Lösung:

```
direktverbindung(A,B) :-
     haelt_in(Nr, A, _, _, _),
     haelt_in(Nr, B, _, _, _).
```

4 (b) Herr Meier möchte abends nach 18:00 Uhr so schnell wie möglich raus aus Darmstadt. Schreiben Sie eine Relation `fahrplan(Abfahrtszeit,Bahnsteig)`, die einen Fahrplan aller ICE oder IC Züge angibt, die nach 18:00 Uhr von Darmstadt abfahren.

Lösung:

```
fahrplan(Abfahrtszeit,Bahnsteig) :-
     zug(Nr,'ICE'),
     haelt_in(Nr, darmstadt, Bahnsteig, _, Abfahrtszeit ),
     Abfahrtszeit > 1800.
fahrplan(Abfahrtszeit,Bahnsteig) :-
     zug(Nr,'IC'),
     haelt_in(Nr, darmstadt, Bahnsteig, _, Abfahrtszeit ),
     Abfahrtszeit > 1800.
```

6

(c) Frau Müller möchte gerne wissen, welche Orte sie von Darmstadt mit genau einmal Umsteigen erreichen kann. Da sie aber gebrechlich ist, interessieren sie nur solche Verbindungen, wo sie beim Umsteigen nicht auf einen anderen Bahnsteig wechseln muß. Definieren Sie in Datalog eine Relation `fahrten(Abfahrtszeit,Umsteigeort,Ankunftsort)`, die alle diese Fahrten umfaßt.

Lösung:

```
fahrten(Abfahrt,Zwischenhalt,Ankunftsort):-
    haelt_in(Nr1, darmstadt, _, _, Abfahrt),
    haelt_in(Nr1, Zwischenhalt, Bahnsteig, ZwischenAnkunft, _),
    haelt_in(Nr2, Zwischenhalt, Bahnsteig,_, ZwischenAbfahrt),
    haelt_in(Nr2, Ankunftsort, _, ZielAnkunft, _),
    Nr1 \= Nr2,
    ZwischenAnkunft > Abfahrt,
    ZwischenAbfahrt > ZwischenAnkunft,
    ZielAnkunft > ZwischenAbfahrt.
```

6. **Fixpunktsemantik**

5 (a) Mittels des EPP (Elementary Production Principle) wurden die folgenden Fakten erzeugt. Für jede Iteration sind jeweils nur die *neu hinzukommenden* Fakten angegeben. Nach der 3. Iteration ist der Fixpunkt erreicht.

1. Iteration: `r(a,b), r(a,c), r(b,c), r(c,d), r(d,e)`

2. Iteration: `s(a,c), s(a,d), s(b,d), s(c,e)`

3. Iteration: `s(c,a), s(d,a), s(d,b), s(e,c)`

Geben Sie ein Datalog-Programm an, das genau dieses Verhalten hervorruft.

Lösung:

```
r(a,b).  r(a,c).  r(b,c).  r(c,d).  r(d,e)
s(X,Y) :- r(X,Z), r(Z,Y).
s(X,Y) :- s(Y,X).
```

2 (b) Nehmen Sie an, die obigen fünf Fakten für `r/2` und acht Fakten für `s/2` seien gegeben. Zusätzlich sei folgende Relation gegeben:

```
t(A) :- s(A,B), \+ r(A,C).
```

Geben Sie alle Fakten an, die mittels dieser Relation abgeleitet werden können.

Anmerkung: Wie in der Vorlesung bezeichnet `\+` die Negation eines Literals.

Lösung:

```
t(e).
```

4 (c) Welches grundsätzliche Problem gibt es mit Negation und der Fixpunkt-Semantik? Tritt das Problem im in Aufgabe (b) betrachteten Fall auf?

Lösung:

Das Problem mit der Negation ist, daß man nicht feststellen kann, daß ein Faktum nicht gilt, bevor man nicht alle gültigen Fakten dieses Prädikats bewiesen hat. Bei direkt oder indirekt rekursiven Prädikaten beißt sich hier die Katze in den Schwanz.
In obigem Fall ist die Negation unproblematisch, da man zuerst alle gültigen Fakten für `r/2` und `s/2` bestimmen kann, bevor man die Fakten für `t/1` ableitet.

7. **Relational Learning**

Gegeben sei folgende Datenbank:

```
q(a).   r(a,b).  r(c,d).     s(a,c,a).   s(c,a,d).   s(a,d,d).
q(b).   r(b,c).  r(d,a).     s(a,a,b).   s(d,d,c).   s(a,c,d).
                             s(a,b,c).   s(c,c,a).   s(c,a,c).
```

positive Beispiele:

```
p(a,b).
p(a,c).
p(b,d).
```

negative Beispiele:

```
p(a,d).
p(c,a).
p(d,c).
```

3 (a) Geben Sie alle Literale an, für die beim Lernen einer nicht-rekursiven Definition für das Zielkonzept `p(X,Y)` im ersten Schritt der *Gain* berechnet werden muß, wenn man annimmt, daß keine neue Variable eingeführt werden darf.

Lösung:

```
q(X), q(Y), r(X,Y), r(Y,X), r(X,X), r(Y,Y),
s(X,X,X), s(X,X,Y), s(X,Y,X), s(Y,X,X),
s(Y,Y,X), s(X,Y,Y), s(Y,X,Y), s(Y,Y,Y)
```

6 (b) In der Vorlesung haben wir kennen gelernt, wie der Algorithmus Dinus dieses Problem in eine Datentabelle transferiert. Führen Sie diese Tranformation für die angegebenen Prädikate durch, wenn das Zielkonzept `p(X,Y)` gelernt werden soll (`Z` ist eine neue Variable).

Lösung:

	`q(Y)`	`r(Y,X)`	`r(Y,Y)`	`r(X,Z)`	`s(X,X,Y)`
a,b	true	false	false	b	true
a,c	false	false	false	b	false
b,d	false	false	false	c	false
a,d	false	true	false	b	false
c,a	true	false	false	d	true
d,c	false	true	false	a	true

3 (c) Warum kann das Literal `s(X,Y,Z)` nicht mit diesem Verfahren behandelt werden?

Lösung:

Es ist nicht determinate, da es für `s(c,a,Z)` und `s(a,c,Z)` jeweils 2 Belegungen für die neue Variable `Z` gibt.

3 (d) Geben Sie alle positiven und negativen Tupel `X,Y,Z` an, die von der Regel `p(X,Y) :- s(X,Z,Y).` abgedeckt werden.

Lösung:

```
a,b,a,+   a,d,d,-   d,c,d,-
a,c,b,+   a,d,c,-   c,a,c,-
```

3 (e) Was versteht man unter Overfitting, und was unternehmen Regel-Lerner wie FOIL dagegen? (1-2 Sätze, ohne Formeln)

Lösung:

Unter Overfitting versteht man eine Überanpassung an die Daten. Regel-Lerner bekämpfen das Problem, indem Sie in gegebenfalls Regeln lernen, die nicht nur positive Beispiele abdecken, bzw. auch positive Beispiele unabgedeckt lassen (Stichwort Pruning).

8. **Semantic Web**

4 (a) Erklären Sie in ein oder zwei Sätzen die unterschiedlichen Ansätze von *Web Mining* und des *Semantic Web* zur Verbesserung des Zugriffs auf die vielfältigen Informationen des Webs.

Lösung:

Web Mining versucht, die Zugriffsmethoden auf das bestehende Web durch den Einsatz vom Methoden des maschinellen Lernens und des Data Minings intelligenter zu machen.
Die Idee des Semantischen Webs ist, Inhalte auch in maschinenlesbarer Form zur Verfügung zu stellen, sodaß Maschinen formale Aussagen interpretieren und daraus Schlüsse ziehen können.

4 (b) Nennen Sie zwei Beispiele für Sachverhalte, die in OWL aber nicht in RDF Schema modelliert werden können.

Lösung:

Zum Beispiel (Folie 49):

- unterschiedliche Bereichseinschränkungen für einzelne Klassen
- Quantoren und Kardinalitätseinschränkungen
- Eigenschaften von Properties (z.B. Transitivität)
- komplexe Mengen-Operationen auf Klassen