# Aufgabenblatt III Künstliche Intelligenz

## Aufgabe 1

Entscheiden Sie, ob die folgenden prädikatenlogischen Formeln korrekte

Formalisierungen des dazugehörigen natürlichsprachlichen Satzes sind:

a) Es gibt keine zwei Fahrzeuge, die dasselbe Nummernschild besitzen.



Antwort 1a

* Wenn x, y und n unterschiedlich sind ist die Aussage richtig. Hier ist die Wissensbasis hilfreich.

b) Das Fahrzeug jeder Person hat 4 Reifen.



Antwort 1b

* Wenn eine Person ein Auto hat, hat das Auto vier Reifen. Diese Aussage ist korrekt.

## Aufgabe 2

Stellen Sie die folgenden Sachverhalte in Prädikatenlogik erster Stufe dar. Achten Sie

auch auf die Repräsentation von nicht explizit erwähnten Sachverhalten, so dass

abgeleitet werden kann, dass Paul die Katze geärgert hat:

- Peter besitzt einen Hund

- Jeder Hundebesitzer ist ein Tierfreund

- Tierfreunde ärgern keine Tiere

- Paul oder Peter haben die Katze geärgert.

Antwort 2

* Person(Peter); Person(Paul)
* Tier(Hund); Tier(Katze)
* Hund :- Person(Peter)
* Tierfreund:- (Person ∧ Hund)
* Tierärgern:- Tierfreund
* Tierägern(Katze):- Paul v Peter

## Aufgabe 3

Es existiert die folgende Knowledge Base:

wizard(ron).

hasWand(harry).

quidditchPlayer(harry).

wizard(X) :- hasBroom(X), hasWand(X).

hasBroom(X) :- quidditchPlayer(X).

Was antwortet PROLOG auf folgende Queries (Anfragen)?

Antwort 3

1. wizard(ron).
   1. true
2. witch(ron).
   1. procedure `witch(A)' does not exist
3. wizard(hermione).
   1. false
4. witch(hermione).
   1. procedure `witch(A)' does not exist
5. wizard(harry).
   1. true
6. wizard(Y).
   1. Y=ron
   2. Y=harry
7. witch(Y).
   1. procedure `witch(A)' does not exist

## Aufgabe 4

Betrachten Sie das folgende Rätsel:

ANNE SAGT: “BETTINA LÜGT”

BETTINA SAGT: “CLAUDIA LÜGT”

CLAUDIA SAGT: “ANNE UND BETTINA LÜGEN”

Antwort 4

1. Finden Sie heraus, wer lügt.
   1. Keine
2. Finden Sie es mit Hilfe des Resolutionsverfahrens heraus.
   1. Bettina
3. Lösen Sie die Aufgabe mit Hilfe von Prolog.
   1. Alle lügen.

**Wissensbasis**:

ver(anne,bettina).

ver(bettina,claudia).

ver(claudia,bettina).

ver(claudia,anne).

luegt(X,Y):- ver(X,Y).

**Abfrage:**

luegt(Y)

## Aufgabe 5

Betrachten Sie folgende Knowledge Base:

bigger(cat,mouse).

bigger(dog,cat).

bigger(sheep,dog).

bigger(horse,sheep).

bigger(elephant,horse).

Schreiben Sie ein rekursives Prädikat biggerThan welches alle “Größer-Als”-Beziehungen

zwischen den Tieren herausfindet (z. B. dass ein Elefant größer ist als eine Maus).

Antwort 5

* biggerThan(X,Y):-bigger(X,Y).
* biggerThan(X,Y):-bigger(X,Z), biggerThan(Z,Y).

## Aufgabe 6

Aus den ersten Semestern dürfte Ihnen die folgende Definition der natürlichen Zahlen

bekannt sein:

1. 0 ist eine Zahl

2. Wenn X eine Zahl ist, dann auch succ(X).

Succ steht für Nachfolger, also ist succ(X) die Zahl, die man erhält, wenn man zu X Eins

addiert. Diese Definition wird zu einem PROLOG-Programm mit folgender Knowledge Base:

zahl(0).

zahl(succ(X)):-zahl(X).

1. Was passiert, wenn die Query „?- zahl(X).“ gestartet wird?

Antwort 6a

* 1. 0 wird ausgegeben, da die Zahl Null eingetragen ist,
  2. Danach „succ(0)“, dann succ(succ(0)) und so weiter, da ein rekursiver Aufruf ist.
     1. Dies kann bis zu einem „sehr lang (unendlich)“ laufenden Programm führen, wenn alle (unendlich) ausgeben lassen möchten

1. Was passiert z.B., wenn wir obige Knowledge Base in folgender Form schreiben: zahl(succ(X)):-zahl(X). zahl(0).

und dann die Query „?- zahl(X).“ starten?

Antwort 6b

* Ist eine Endlos schleife, da die Basisklausel nicht am Anfang steht. Das Programm läuft auf einen Fehler. Es wird das Stack limit erreicht.

## Aufgabe 7

Betrachten Sie die folgende Knowledge Base:

directTrain(lueneburg,hamburg).

directTrain(bremen,lueneburg).

directTrain(braunschweig,hannover).

directTrain(hannover,lueneburg).

directTrain(hamburg,kiel).

directTrain(wolfenbuettel,braunschweig).

directTrain(berlin,braunschweig).

Die Knowledge Base enthält Informationen über direkte Zugverbindungen. Natürlich kann

man auch reisen, indem man mehrere Züge hintereinander nimmt.

a) Schreiben Sie ein rekursives Prädikat travelBetween, das angibt, ob man zwischen zwei

Orten fahren kann (bspw. sollte die Query: „travelBetween(berlin,hamburg).“ 'yes'

ergeben).

Antwort 7a

* travelBetween(X,Y):-directTrain(X,Y).
* travelBetween(X,Y):-directTrain(X,Z), travelBetween(Z,Y).

b) Gehen Sie davon aus, dass eine Verbindung von A nach B auch eine Verbindung von B

nach A bedeutet. Ergänzen Sie eine entsprechende Regel und testen Sie diese mit der

Query: travelBetween(hamburg,berlin).

Antwort 7b

* directTrain(X,Y):- directTrain(Y,X).
* erzeugt aber eine Endlosschleife, das Programm kommt zu keinem Ergebniss.

## Aufgabe 8

Betrachten Sie die folgende Knowledge Base:

byCar(sydney,melbourne).

byCar(melbourne,canberra).

byCar(nancy,saarbruecken).

byCar(nancy,metz).

byTrain(metz,frankfurt).

byTrain(saarbruecken,frankfurt).

byTrain(metz,paris).

byTrain(saarbruecken,paris).

byPlane(frankfurt,singapore).

byPlane(paris,losAngeles).

byPlane(singapore,sydney).

byPlane(losAngeles,sydney).

Schreiben Sie ein Prädikat travel, das angibt, ob es möglich ist von einem Ort zum

anderen zu reisen, indem man Auto, Zug oder Flugzeug nimmt. Zum Beispiel sollte 'yes'

auf die Query travel(nancy,canberra) geliefert werden.

Antwort 8

* travel(X,Y):- byTrain(X,Y); byCar(X,Y); byPlane(X,Y).
* travel(X,Y):- (byTrain(X,Z); byCar(X,Z); byPlane(X,Z)), travel(Z,Y).

## Aufgabe 9

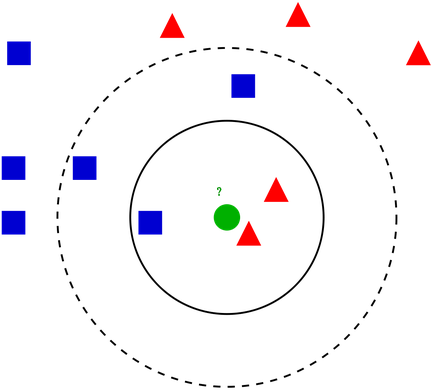
Beschreiben Sie die prinzipielle Funktionsweise des Nearest-Neighbor-Verfahrens.

Verwenden Sie zur Erläuterung ein selbst gewähltes, einfaches Beispiel. Nennen Sie

mögliche Probleme bei der Anwendung des Verfahrens.

Antwort 9

* Bei dem Nearest-Neighbor-Verfahren wird die Klassifizierung eines Unbekannten Datenpunktes anhand der nächst gelegenen Datenpunkte vorgenommen.
* Die Herausforderung ist, den Betrachtungsraum passend zu wählen. Wird dieser zu klein gewählt findet keine Generalisierung statt. Wired dieser zu groß gewählt, kann das Verfahren zu unspezifisch werden.
* Bei folgendem Beispiel ist die Herausforderung zu sehen. Eine Möglichkeit ist, die Abstände relativ zu betrachten und somit die Gewichtung der Nachbarn zu bewerten.



## Aufgabe 10

Beschreiben Sie die prinzipielle Funktionsweise von Entscheidungsbäumen anhand des

folgenden Beispiels. Verwenden Sie den Informationsgewinn als Maß für die Qualität des

Splittings. Berechnen Sie den vollständigen Entscheidungsbaum.

Antwort 10

Aussicht

sonnig

regnerisch

bedeckt

Nein

Temperatur

Ja

warm

kalt

Nein

Wind

schwach

stark

Nein

Ja

Entropie = -(5/10\*log2(5/10))- (5/10\*log2(5/10)) = 1

Aussicht: Informationsgewinn = 0,725 = 1-4/10\*0-3/10\*0,918-3/10\*0

* sonnig T = 0 = -(4/4\*log2(4/4))
* bedeckt T = 0,918 = -(1/3\*log2(1/3))-(2/3\*log2(2/3))
* regnerisch T = 0

Temperatur: Informationsgewinn = 0125 = 1-6/10\*0,918-4/10\*0,811

* warm T = 0,918
* kalt T = 0,811

Wind: Informationsgewinn = 0,0351 = 1-3/10\*0,918-7/10\*0,985

* stark T = 0,985
* schwach T = 0,918