

**Technische Universität
München**

Fakultät für Informatik

Forschungs- und Lehrereinheit Informatik VI

Symbolisches Lernen

Proseminar Kognitive Robotik (SS12)

Johannes Klein

Betreuer: Dr. Florian Röhrbein

Leitung: Prof. Alois Knoll

Abgabetermin: 21. Juli 2012

Inhaltsverzeichnis

1	Einführung und Überblick	3
2	Symbolisches Lernen	3
2.1	Arten symbolischer Lernverfahren	4
2.1.1	Induktive Lernverfahren	4
2.1.2	Deduktive Lernverfahren	4
2.1.3	Mischformen	5
2.2	Formen der Wissensrepräsentation	6
3	Induktion von Entscheidungsbäumen	7
3.1	Ein Objektbeispiel	7
3.2	Der Trainingsdatensatz	8
3.3	Konstruktion von Entscheidungsbäumen	8
3.3.1	Der Algorithmus in Pseudo-Code	9
3.3.2	Erklärung	9
4	ID3 - ein TDIDT Algorithmus	10
4.1	Effektives Finden eines Entscheidungsbaumes	11
4.2	Konstruktion einfacher Entscheidungsbäume	11
4.3	Ein durch ID3 konstruierter Entscheidungsbaum	12
	Literaturverzeichnis	15

1 Einführung und Überblick

Diese Arbeit handelt über symbolisches Lernen. Dabei bedeutet Lernen die Konstruktion oder Modifikation von Repräsentationen dessen, was erfahren wird.¹ Darunter wird verstanden, dass aus einem Universum, bestehend aus Objekten, Strukturen ermittelt werden können, welche man mithilfe von Wissensmodellierung darstellen kann. Zur Wissensrepräsentation verwende ich in dieser Arbeit den Entscheidungsbaum.

Subsymbolische Verfahren, die auf neuronale Netze basieren, sind nicht Bestandteil dieser Arbeit. Denn während „die symbolischen Verfahren auf explizitem Wissen operieren, das auch zur Kommunikation mit dem Benutzer verwendet werden kann, manipulieren subsymbolische Verfahren im wesentlichen Informationen, die für den Benutzer keine explizite Bedeutung haben (Gewichte in einem Netz, Übergangswahrscheinlichkeiten in einem Automaten, etc.)“². Dies widerspricht dem Begriff der symbolischen Lernverfahren, weil „explizite Symbole manipuliert werden“³, darum behandle ich sie in dieser Arbeit nicht.

Im folgenden Kapitel werden verschiedene Arten von symbolischen Lernverfahren dargestellt. Anschließend stelle ich Ihnen im Kapitel 3 die Wissensrepräsentation mithilfe von Entscheidungsbäumen genauer vor. Gefolgt wird dies durch Kapitel 4, in welchem ich einen Algorithmus zur Entscheidungsbaumsynthese vorstelle. Dieser zeichnet sich dadurch aus, dass er effektiv einfache Entscheidungsbäume findet.

2 Symbolisches Lernen

Wenn man allgemein von Lernen spricht, dann umfasst dies ein sehr breites Spektrum. Zum einen gibt es die Möglichkeit neues Wissen aufzunehmen, wie beispielsweise ein Roboter, der in neues Terrain vorstößt und seine Karte erweitert, zum anderen kann man erworbenes Wissen analysieren, um neue Strukturen zu entdecken. Dies könnte ein Bewegungsroboter umsetzen, indem er bisherige Routen untersucht, um effektivere Wege zu finden. Schließlich können noch neue Fertigkeiten erlernt werden, sowohl motorisch als auch kognitiv, welche dem System helfen, auch bisher unbekannte Situationen besser zu meistern. Da es sich hierbei um einen sehr komplexen Bereich handelt, ist es nicht verwunderlich, dass mehrere Lernverfahren entwickelt wurden, aus denen ich Ihnen einen Ausschnitt vorstellen möchte.

¹MICHALSKI/Vgl. S. 10.

²JÜRGEN/S. 25.

³JÜRGEN/S. 24.

2.1 Arten symbolischer Lernverfahren

Um Lernverfahren einordnen zu können, beschreibt Kreuziger⁴, die von Michalski⁵ entwickelte Multikriterien-Klassifikation für Lernverfahren, welche in Abbildung 1 dargestellt wird. Danach unterscheidet Michalski „nach dem Zweck des Lernens, dem Typ der Eingabeinformation, der verwendeten Inferenzmethode und der Rolle des vorhandenen Hintergrundwissens.“⁶

Die folgenden Unterpunkte unterscheiden sich anhand ihrer gewählten Inferenzmethode.

2.1.1 Induktive Lernverfahren

Induktive Verfahren werden hauptsächlich beim Begriffs- und Regellernen eingesetzt. Vorausgesetzt wird eine große Menge geeigneter Beispiele, woraus versucht wird „eine möglichst plausible allgemeingültige Aussage abzuleiten, die als induktive Hypothese bezeichnet wird.“⁷ Kreuziger erwähnt auch, dass induktive Lernverfahren auch synthetisch genannt werden, „weil sie - im Gegensatz zu deduktiven Verfahren - tatsächlich neues Wissen aufbauen, d.h. Wissen, das nicht durch streng logische Deduktion aus dem vorhandenen Wissen abgeleitet werden kann.“⁸

Die induktiven Lernverfahren kann man ebenfalls in mehrere Subkategorien unterteilen, wie aus Abbildung 2 entnommen werden kann. Dies würde aber den Rahmen dieser Seminararbeit sprengen und wurde deshalb nicht mitaufgenommen.

2.1.2 Deduktive Lernverfahren

„Bei den deduktiven oder analytischen Lernverfahren wird versucht, bereits vorhandenes Wissen zu analysieren und in eine effektivere oder „operationalere“, d.h. direkt vom System verwendbare, Form zu bringen.“⁹ Außerdem erwähnt Kreuziger¹⁰, dass durch eine gerechtfertigte Generalisierung die Leistungsfähigkeit des Systems verbessert werden kann und dass diese Lernmethode hauptsächlich dazu verwendet wird, um ein schnelleres Lernen zu ermöglichen. Eine außerordentlich wichtige Eigenschaft dieses Lernverfahrens ist auch, dass deduktive „Lernverfahren wahrheitserhaltend [sind], d.h. aus bestehendem wahren Wissen wird durch deduktive Lernverfahren wieder wahres Wissen abgeleitet.“¹¹

⁴JÜRGEN/Vgl. S. 24.

⁵MICHALSKI/KODRATOFF/Vgl. S. 15.

⁶JÜRGEN/S. 24.

⁷JÜRGEN/S. 24.

⁸JÜRGEN/S. 29.

⁹JÜRGEN/S. 36.

¹⁰JÜRGEN/Vgl. S. 36.

¹¹JÜRGEN/S. 36.

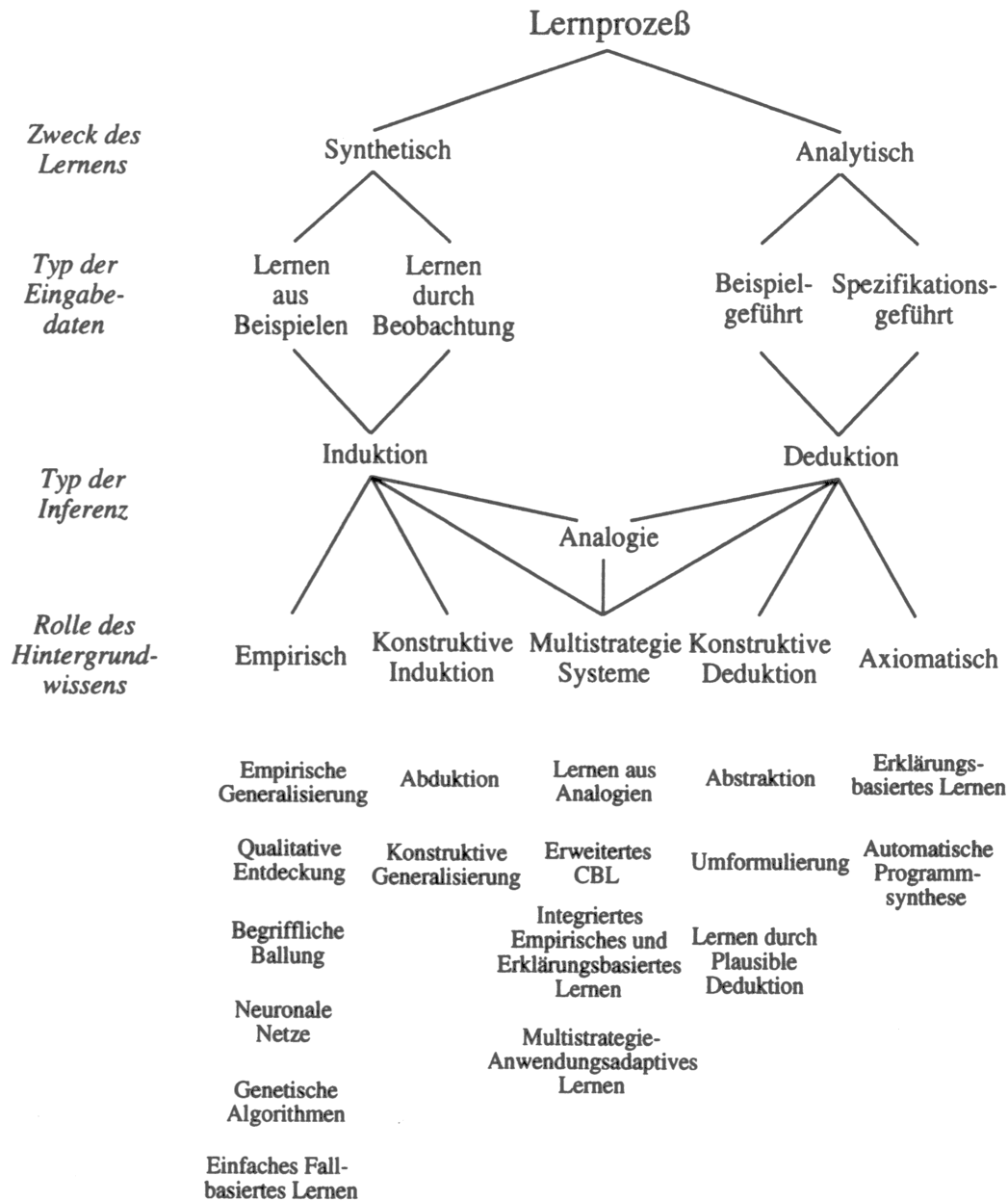


Abbildung 1: Multikriterien-Klassifikation nach Michalski: JÜRGEN/S. 25,

2.1.3 Mischformen

Dies sind Verfahren, die Komponenten der Induktion und Deduktion beinhalten. Als Beispiel nenne ich das Lernen aus Analogien. „Dabei soll eine Lösung für ein neues Problem oder eine Beschreibung für einen neuen Begriff dadurch gefunden werden, daß in einer Wissensbasis bereits bekannter Lösungen bzw. Begriffe ein ähnlicher Fall gesucht wird und

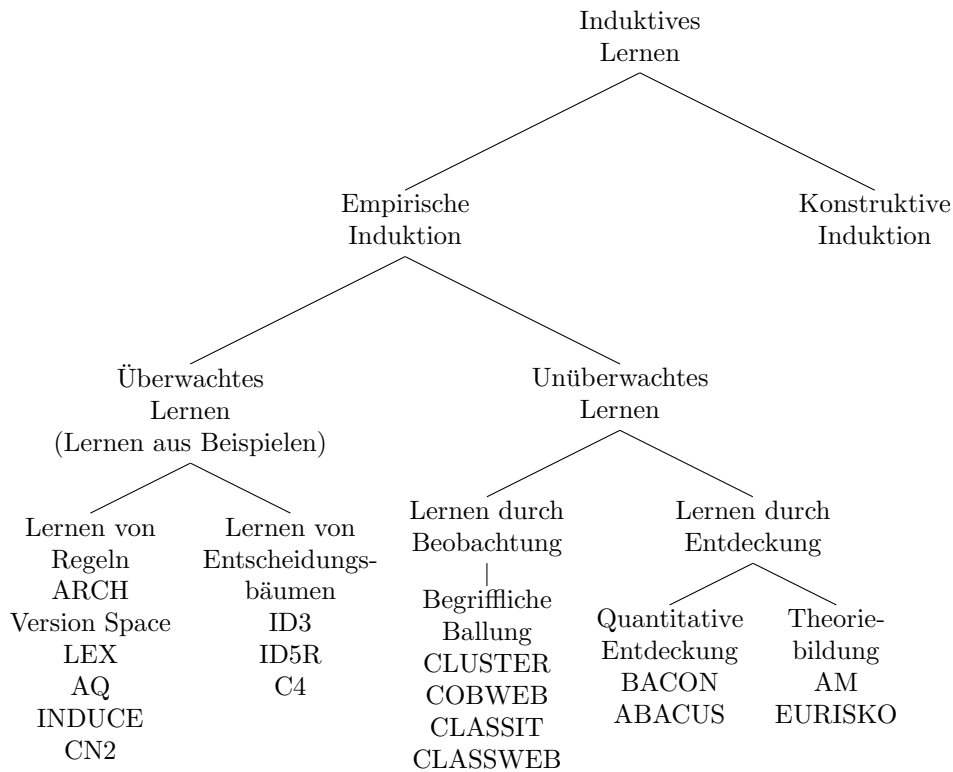


Abbildung 2: Darstellung induktiver Lernverfahren: JÜRGEN/S.31,

die Informationen darüber auf den neuen Fall übertragen werden.“¹²

2.2 Formen der Wissensrepräsentation

Da, wie bereits erwähnt, symbolische Verfahren auf explizitem Wissen operieren, kann dies auch formal abgebildet werden. Kreuziger listet in seinem Werk¹³ folgende Formen auf:

- Einfache Attribut-Wert-Paare: Beispiele werden durch eine feste Anzahl von Attributen und deren konkreter Wertbelegung repräsentiert. Die Werte können nominal oder numerisch sein.
- Aussagen- oder prädikatenlogische Ausdrücke: Ein Beispiel ist durch einen entsprechenden Ausdruck beschrieben. Das Resultat des Lernvorgangs ist entsprechend dargestellt.
- Regeln: Ergebnisse eines Lernschritts können als Klassifikationsregel dargestellt werden. Außerdem werden Regeln zur Darstellung von Planungsoperatoren verwendet.

¹²JÜRGEN/S. 38.

¹³JÜRGEN/S. 29.

- Entscheidungsbäume: Das Resultat eines Lernvorgangs wird als Baum dargestellt mit inneren Entscheidungs- und äußeren Klassifikationsknoten.
- Semantische Netze: Beispiele für den Lernvorgang oder Resultate sind beschrieben durch eine relationale Struktur.

Falls Sie zur Thematik der Wissensrepräsentation mehr erfahren möchten, so empfehle ich Ihnen das Buch *Einführung in die künstliche Intelligenz* von Günther Görz¹⁴. Darin beschreibt der Autor im Kapitel 1 *Wissensrepräsentation und Logik* Aufgaben, Modellierung, Repräsentationsformalismen, etc. anhand prädikatenlogischer Ausdrücke.

3 Induktion von Entscheidungsbäumen

In diesem Kapitel wird die Induktionsaufgabe beschrieben, welche die Grundlage des symbolischen Lernens durch Entscheidungsbäume bildet. Dabei wird das Resultat eines Lernvorgangs „als Baum dargestellt mit inneren Entscheidungs- und äußeren Klassifikationsknoten.“¹⁵ Nach Quinlan¹⁶ wird als Basis von einem Universum, bestehend aus Objekten, ausgegangen, wobei jedes Objekt ausdrücklich mit einer Sammlung von Attributen beschrieben ist. Außerdem misst jede dieser Eigenschaften mehrere wichtige Abstraktionen eines Objekts und beschränkt diese auf einen Satz diskreter Werte. „Ziel ist eine Struktur, [welche dem Ergebnis der Lernaufgabe entspricht,] die eine Einordnung eines Beispiels in eine der Klassen zuläßt“¹⁷

3.1 Ein Objektbeispiel

Betrachtet man beispielsweise das Objekt **Tier erlegt Beute** und die Klassifikationsaufgabe besteht darin zu ermitteln, ob das Tier Erfolg hat, dann könnten die Attribute wie folgt aussehen:

- Das Verhalten der Beute,
mit den Werten {*flüchtet, versteckt sich, greift an*}
- Der Stärkeunterschied der Beute im Vergleich zum Angreifer,
mit den Werten {*schwächer, gleichstark, stärker*}
- Der Geschwindigkeitsunterschied des flüchtenden Tieres
im Vergleich zum angreifenden Tier,
mit den Werten {*schneller, langsamer*}

¹⁴GÖRZ.

¹⁵JÜRGEN/S. 29.

¹⁶QUINLAN/Vgl. S. 351.

¹⁷JÜRGEN/S. 33.

- Die Beute hat eine besondere Fähigkeit wie fliegen, schwimmen, etc. mit den Werten $\{wahr, falsch\}$

Anmerkung: In der Realität müssen Attributwerte sehr klar definiert werden, um sie messen, vergleichen, gewichten, etc. zu können. Um dieses Beispiel kurz und einfach zu halten, wurde bewusst auf Konkretisierung verzichtet.

3.2 Der Trainingsdatensatz

Die Aufgabe der Induktion besteht darin, Klassifizierungsregeln zu entwickeln, welche die Klassen von jedem benötigtem Objekt bestimmen können. Dazu ist es unerlässlich, dass zum Erstellen des Entscheidungsbaumes bereits ein Trainingssatz von Objekten mit zugehöriger Klasse bekannt ist. Zwingendermaßen muss der Datensatz eine hohe Güte aufweisen, da sich Fehler im Baum aufaddieren und man somit an Genauigkeit verliert. Zudem nennt Quinlan¹⁸ den Fall, dass zwei Objekte mit exakt gleicher Attributbelegung, aber mit unterschiedlicher Klassenzugehörigkeit, nicht klar unterschieden werden können. In diesem Fall sind alleinig die Attribute der Objekte unzureichend für den Trainingsdatensatz. Außerdem muss der Datensatz hinreichend viele Objekte enthalten, sodass jede mögliche Klasse auch tatsächlich durch mindestens ein Objekt im Datensatz vertreten wird. Ist dies nicht gegeben, so könnte der Extremfall auftreten, dass der Datensatz nur aus Objekten einer einzigen Klasse besteht. Dann besteht der Baum lediglich aus einem Blatt und alle Objekte mit abweichender Klassenzugehörigkeit können nicht zugeordnet werden.

Abbildung 3 zeigt einen stark vereinfachten Trainingssatz, welcher für das Beispiel **Tier erlegt Beute** benutzt werden kann. Jedes der 14 enthaltenen Objekten ist mit der jeweiligen Objektklasse (P für positiv und N für negativ) und den jeweiligen Objektwerten gelistet.

3.3 Konstruktion von Entscheidungsbäumen

Das Wesentliche des Verallgemeinens ist es aber auch, dass bisher unbekannte Objekte, jenseits des Trainingsdatensatzes, korrekt erkannt und der entsprechenden Klasse zugewiesen werden können. Um dies zu bewerkstelligen, muss der Entscheidungsbaum einige bedeutende Beziehungen zwischen Objektklassen und deren Attributwerten herstellen können. Wie Entscheidungsbäume aus Beispielen konstruiert werden können, erfahren Sie im folgenden Verfahren. Dazu nennt Grieser¹⁹ die Methode der *Top Down Induction of Decision Trees* (kurz TDIDT), welche auf eine Arbeit aus den sechziger Jahren zurückgeht.

¹⁸QUINLAN/Vgl. S. 351.

¹⁹GRIESER/Vgl. S. 105.

No.	Attribute				Klasse
	Verhalten	Stärke	Geschwindigkeit	bes. Fähigkeit	
1	flüchtet	schwächer	langsamer	wahr	P
2	flüchtet	schwächer	langsamer	falsch	P
3	versteckt sich	schwächer	langsamer	wahr	N
4	greift an	gleichstark	langsamer	wahr	N
5	greift an	stärker	schneller	wahr	N
6	greift an	schwächer	schneller	falsch	P
7	versteckt sich	stärker	schneller	falsch	N
8	flüchtet	gleichstark	langsamer	wahr	P
9	flüchtet	stärker	schneller	wahr	N
10	greift an	gleichstark	schneller	wahr	N
11	flüchtet	gleichstark	schneller	falsch	N
12	versteckt sich	gleichstark	langsamer	falsch	N
13	versteckt sich	schwächer	schneller	wahr	N
14	greift an	gleichstark	langsamer	falsch	P

Abbildung 3: Ein Beispieltrainingsdatensatz für die Klassifikation von **Tier erlegt Beute**

3.3.1 Der Algorithmus in Pseudo-Code

```

function INDUCTION-OF-DT(attribut, trainingsdatensatz)
  if alle Objekte des trainingsdatensatz haben die gleiche Klasse then
    ▷ beendet die Rekursion und gibt die entsprechende Klasse zurück
    return erstellt ein Blatt mit dem zugehörigen Klassennamen;
  else
    ▷ erstellt Teilbaum aus der entsprechenden Teilmenge
    ▷ wählt ein bisher unbenutztes Attribut als Wurzel des neuen Teilbaums
    INDUCTION-OF-DT(bisher unbenutztes attribut, Teilmenge.trainingsdatensatz);
  end if
end function

```

3.3.2 Erklärung

Durch das Top-Down-Verfahren beginnt jede Konstruktion eines Entscheidungsbaums bei der Wurzel. Ausgehend davon wird für jeden Ausgang aus der Wurzel, also für die Anzahl aller verschiedenen Attributwerte, rekursiv ein Teilbaum konstruiert. Dabei wird die Beispielmenge bezüglich der Ausgänge der Tests partitioniert und durch die beschriebene Methode rekursiv auf jede Beispielpartition angewendet. Dieser Prozess endet, wenn die jeweilige Beispielmenge nur noch gleich klassifizierte Elemente enthält.²⁰

²⁰GRIESER/Vgl. S. 105.

Ein so konstruierter Entscheidungsbaum wird in Abbildung 4 dargestellt.

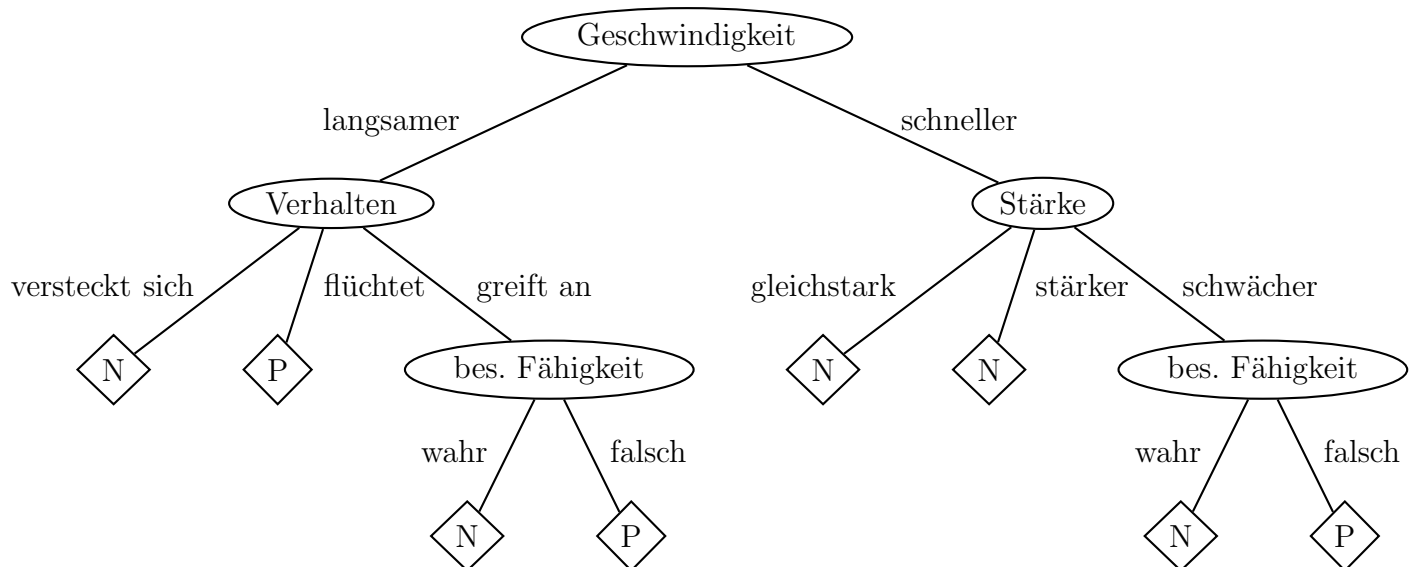


Abbildung 4: Dieser Entscheidungsbaum wurde mithilfe von Induktion aus den bereits bekannten Daten aus Abbildung 3 erstellt.

Ein bekannter Algorithmus zur Entscheidungsbaumfindung ist der ID3. Dieser baut auf dem Algorithmus CART auf und dient selbst als Grundlage des C4.5 Algorithmus. Dies sind Vertreter der TDIDT Familie und unterscheiden sich unter anderem durch die Auswahl der einzelnen Knoten. Da wir jetzt die Grundlage zur Synthese eines Entscheidungsbaumes kennen, stelle ich Ihnen im folgenden Kapitel den Algorithmus ID3 vor.

4 ID3 - ein TDIDT Algorithmus

Wird das obig genannte Verfahren angewendet, dann erhält man zwar stets einen Entscheidungsbaum, welcher alle Elemente der Trainingsmenge korrekt klassifiziert. Doch sind die entstehenden Bäume oft weit komplexer als nötig. Sind die Daten zusätzlich auch nur leicht verrauscht oder decken den Lernbereich nicht oder ungleichmäßig ab, so korrespondiert die Struktur der entstehenden Bäumen nicht mit der des zu lernenden Konzeptes, und der Generalisierungsfehler steigt trotz des minimalisierten Trainingsfehlers an.²¹ Deshalb sind möglichst einfache Entscheidungsbäume zu bevorzugen, um den Generalisierungsfehler möglichst klein zu halten. Da es sehr aufwendig ist aus dem Trainingssatz alle Entscheidungsbäume zu generieren und den einfachsten auszuwählen, wurden diverse Algorithmen entwickelt.

²¹GRIESER/Vgl. S. 105.

Der in diesem Kapitel vorgestellte Algorithmus ID3, hat das Ziel, effizient einfache Bäume zu finden. Dieser kann aber aufgrund seiner iterativen Grundstruktur nicht garantieren, dass sich der konstruierte Baum an den finalen Entscheidungsbaum annähert, sofern nicht der gesamte Datensatz betrachtet wird, wie O’Keefe²² anmerkte. Im Anschluss zeige ich Ihnen zwei Methoden, mit denen der ID3 schnell einen Entscheidungsbaum findet und wodurch die Tiefe und Spannweite des Baumes reduziert wird.

4.1 Effektives Finden eines Entscheidungsbaumes

Um einen Entscheidungsbaum effektiv zu finden, nennt Quinlan²³ eine Vorgehensweise, bei der der Algorithmus eine zufällige Teilmenge mit allen zugehörigen Objekten aus dem Trainingsdatensatz wählt und daraus einen Entscheidungsbaum erstellt. Anschließend wird der Baum anhand der nicht gewählten Objekte getestet. Werden dabei Objekte nicht korrekt klassifiziert, so wird davon ein Teil ausgewählt und in die zuvor gewählte Teilmenge mitaufgenommen. Dann wird der Baum abermals getestet. Dieser iterative Prozess wird so lange wiederholt, bis mit dem gefundenen Entscheidungsbaum alle Objekte korrekt zugeordnet werden. Dazugehörig beruft sich Quinlan²⁴ auf empirische Studien, welche besagen, dass auf diese Weise ein korrekter Entscheidungsbaum gewöhnlich viel schneller gefunden wird, anstelle den Entscheidungsbaum direkt aus dem gesamten Datensatz zu formen.

4.2 Konstruktion einfacher Entscheidungsbäume

Möchte man möglichst einfache Entscheidungsbäume erhalten, so kann man als mathematische Grundlage die Informationstheorie nach Shannon²⁵ anwenden. Da Entscheidungsbäume Information über die Klassifikation der im Trainingsdatensatz enthaltener Beispiele vermitteln, lässt sich der Informationsgehalt eines Baums aus den Wahrscheinlichkeiten der verschiedenen Klassifikationen berechnen.²⁶ Dazu verwendet man die Formel

$$I(c_i) = \sum_{i \in \text{Klasse}} \left(-\frac{c_i}{c} \log_2 \frac{c_i}{c} \right),$$

welche mit der Einheit $[I] = \text{bit}$ angegeben wird.

Dabei entspricht $c = c_1 + c_2 + \dots + c_i$, also der Gesamtanzahl aller Objekte. c_i repräsentiert die Anzahl der Objekte, welche in der jeweiligen Klasse enthalten sind.

²²O’KEEFE/Vgl. S. 480.

²³QUINLAN/Vgl. S. 352.

²⁴QUINLAN/Vgl. S. 352.

²⁵SHANNON.

²⁶LUGER/Vgl. S. 418.

Angewendet auf den Trainingsdatensatz aus Abbildung 3 erhält man mit $p = c_1 = 5$, $n = c_2 = 9$ und $a = 2$, da die Klassen P und N verwendet werden,

$$\begin{aligned} I(p, n) &= -\frac{5}{14} \log_2 \frac{5}{14} - \frac{9}{14} \log_2 \frac{9}{14} \\ &= 0,940 \text{ bit} \end{aligned}$$

als Informationsgehalt für den vollständigen Entscheidungsbaum.

Möchte man den Informationsgehalt eines einzelnen Attributes berechnen, so verwendet man

$$E(A) = \sum_{i=1}^v \left(-\frac{c_i}{c} I(c_i)\right), c_i \in \text{Klasse},$$

um zu berechnen, welche Informationsmenge zur Fertigstellung des Baumes notwendig ist, wenn das gewählte Attribut an der Wurzel stünde. Dies subtrahiert man von dem anfangs berechnetem Gesamtinformationsgehalt und man erhält den Informationsgewinn durch das verwendete Attribut.

$$\text{gewinn}(\text{Attribut}) = I(\text{Gesamtbaum}) - E(\text{Attribut})$$

4.3 Ein durch ID3 konstruierter Entscheidungsbaum

Durchläuft man das Verfahren des ID3, angewandt auf folgend zufällig ausgewählte Teilmenge des Trainingsdatensatzes, dann erhält man folgende Ergebnisse:

No.	Attribute				Klasse
	Verhalten	Stärke	Geschwindigkeit	bes. Fähigkeit	
1	flüchtet	schwächer	langsamer	wahr	P
3	versteckt sich	schwächer	langsamer	wahr	N
4	greift an	gleichstark	langsamer	wahr	N
5	greift an	stärker	schneller	wahr	N
6	greift an	schwächer	schneller	falsch	P
8	flüchtet	gleichstark	langsamer	wahr	P
11	flüchtet	gleichstark	schneller	falsch	N
13	versteckt sich	schwächer	schneller	wahr	N
14	greift an	gleichstark	langsamer	falsch	P

Abbildung 5: Das gewählte Fenster aus dem bereits bekanntem Datensatz aus Abbildung 3.

Angewendet auf das gewählte Fenster enthält der Baum den Informationsgehalt:

$$\begin{aligned} I(4, 5) &= -\frac{4}{9} \log_2 \frac{4}{9} - \frac{5}{9} \log_2 \frac{5}{9} \\ &= 0,991 \text{ bit} \end{aligned}$$

Attributwerte für die Information aus *Verhalten*:

- Attributwert: flüchtet

$$\begin{aligned} I(2, 1) &= -\frac{2}{3} \log_2 \frac{2}{3} - \frac{1}{3} \log_2 \frac{1}{3} \\ &= 0,918 \text{ bit} \end{aligned}$$

- Attributwert: versteckt sich

$$\begin{aligned} I(0, 2) &= -\frac{0}{2} \log_2 \frac{0}{2} - \frac{2}{2} \log_2 \frac{2}{2} \\ &= 0 \text{ bit} \end{aligned}$$

- Attributwert: greift an

$$\begin{aligned} I(2, 2) &= -\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} \\ &= 1 \text{ bit} \end{aligned}$$

Für $E(\textit{Verhalten})$ erhält man dann:

$$\begin{aligned} E(\textit{Verhalten}) &= \frac{3}{9} I(2, 1) + \frac{2}{9} I(0, 2) + \frac{4}{9} I(2, 2) \\ &= 0,751 \text{ bit} \end{aligned}$$

Daraus folgt der Informationsgewinn für das Attribut *Verhalten*:

$$\begin{aligned} \textit{gewinn}(\textit{Verhalten}) &= I(4, 5) - E(\textit{Verhalten}) \\ &= 0,240 \text{ bit} \end{aligned}$$

Für die anderen Attribute erhält man die Werte analog:

$$\begin{aligned} \textit{gewinn}(\textit{Stärke}) &= 0,102 \text{ bit} \\ \textit{gewinn}(\textit{Geschwindigkeit}) &= 0,091 \text{ bit} \\ \textit{gewinn}(\textit{bes. Fähigkeit}) &= 0,073 \text{ bit} \end{aligned}$$

Somit wählt man das Attribut *Verhalten* als Wurzel für den Entscheidungsbaum und betrachtet die entstandenen Teilmengen. Folgt man dem Ast *versteckt sich* so bemerkt man, dass alle Objekte dieser Teilmenge bereits einer einzigen Klasse angehören. Somit sind diese Objekte klassifiziert und bilden ein Blatt mit der Klasse N . Die Teilmenge des Astes *greift an* wird durch das Attribut *besondere Fähigkeit* am besten klassifiziert. Die beiden

so entstandenen Äste führen über den Attributwert *wahr* zum Klassenblatt *N* bzw. über *falsch* zu *P*. Folgt man dem Ast *flüchtet* des Attributes *Verhalten*, so erhält man für die Teilmenge durch die Attribute *Geschwindigkeit* und *besondere Fähigkeit* den gleichen Informationsgewinn. Da allerdings das Attribut *Geschwindigkeit* bei der Wahl der Wurzel einen höheren Gewinn aufweist und somit anzunehmen ist, dass bei bisher unbekanntem Objekten dies ebenfalls so ist, sofern der Trainingsdatensatz ausgewogen ermittelt wurde, wird *Geschwindigkeit* gewählt. Die Objekte, die nun *langsamer* als Attributwert haben, werden als *P* bzw. diejenigen die *schneller* als Attributwert haben, werden als *N* klassifiziert. Da nun alle Objekte aus dem gewählten Fenster klassifiziert wurden, testet man die restlichen Objekte an dem konstruierten Entscheidungsbaum. Da auch diese durch den Baum korrekt klassifiziert werden, ist der Entscheidungsbaum korrekt und das Verfahren ist abgeschlossen. Der gefundene Baum wird in Abbildung 6 gezeigt und ist wesentlich einfacher aufgebaut als der Baum aus Abbildung 4.

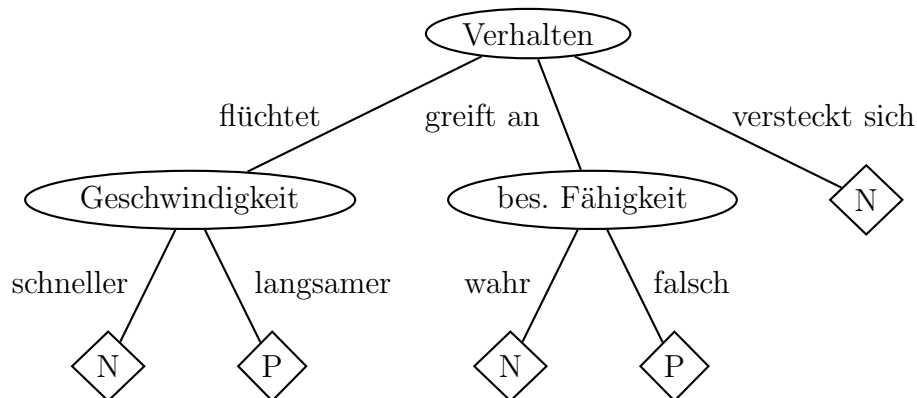


Abbildung 6: Ein möglicher, durch ID3 konstruierter, Entscheidungsbaum.

Literatur

- Grieser, Gunter:** Selbsteinschätzende Lernverfahren: Möglichkeiten und Grenzen. Berlin: Aka, 2002
- Görz, Guenther:** Einführung in die künstliche Intelligenz -. Addison-Wesley, 1995
- Jürgen, Kreuziger:** Eine Architektur zur Anwendung symbolischer Lernverfahren in der Robotik. Infix, 1994
- Luger, George F.:** Künstliche Intelligenz - Strategien zur Lösung komplexer Probleme. Pearson Studium, 2001
- Michalski, R. S.:** Understanding the Nature of Learning: Issues and Research Directions. In **Michalski, R. S./Carbonell, J. G./Mitchell, T. M. (Hrsg.):** Machine Learning: An Artificial Intelligence Approach: Volume II. Los Altos, CA: Kaufmann, 1986, 3–25
- Michalski, R. S./Kodratoff, Y.:** Research in Machine Learning: Recent Progress, Classification of Methods, and Future Directions. In **Kodratoff, Y./Michalski, R. S. (Hrsg.):** Machine Learning: An Artificial Intelligence Approach (Volume III). San Mateo, CA: Kaufmann, 1990, 3–30
- O’Keefe, R.A.:** Concept Formation from Very Large Training Sets. In Eighth International Joint Conference on Artificial Intelligence. Morgan Kaufmann, 1983, 479–481
- Quinlan, J. R.:** Induction of Decision Trees. In **Buchanan, B. G./Wilkins, D. C. (Hrsg.):** Readings in Knowledge Acquisition and Learning: Automating the Construction and Improvement of Expert Systems. San Mateo, CA: Kaufmann, 1993, 349–361
- Shannon, C. E.:** A mathematical theory of communication (part I). BSTJ, 27 1948, 379–423