

2. Aussagenlogik

2.1 Syntax

Aussagenlogik untersucht Verknüpfungen wie "und", "oder", "nicht", "wenn ... dann" zwischen atomaren und komplexen Sätzen.

Sätze selbst sind entweder wahr oder falsch. Ansonsten ist ihre Bedeutung irrelevant.

Atomare Sätze werden durch beliebige Symbole repräsentiert: bei Schöning A_1, A_2, \dots ; oft sind für den Benutzer Symbole bequem, die die intendierte Bedeutung wiedergeben (Regen, Nass, Verwandt), werden bei Schöning als Abkürzung für ein A_i eingeführt.

Nochmal: intendierte Bedeutung für die Logik irrelevant!!

Def. (Syntax der Aussagenlogik)

Sei A eine Menge von Symbolen (genannt aussagenlogische Variablen oder atomare Formeln).

Die Menge der (aussagenlogischen) Formeln über A ist die kleinste Menge, für die gilt:

1. Jede atomare Formel aus A ist eine Formel.
2. Wenn F und G Formeln sind, so sind $(F \wedge G)$ und $(F \vee G)$ Formeln.
3. Wenn F eine Formel ist, so ist $\neg F$ eine Formel.

$\neg F$ heißt Negation von F , $(F \wedge G)$ Konjunktion von F und G ,

$(F \vee G)$ Disjunktion von F und G .

Eine Formel F , die als Teil einer Formel G vorkommt, heißt Teilformel von G .

Abkürzungen:

$(F1 \rightarrow F2)$ statt $(\neg F1 \vee F2)$

$(F1 \leftrightarrow F2)$ statt $((F1 \wedge F2) \vee (\neg F1 \wedge \neg F2))$

Bemerkung: man könnte diese Symbole auch direkt in der Definition der Syntax der Aussagenlogik einführen. Die Einführung als Abkürzung vereinfacht aber Induktionsbeweise über den Aufbau von Formeln.

Zu beachten: Implikation gibt umgangssprachliche Verwendung von "wenn...dann" nur näherungsweise wieder.

1) Wenn Leipzig in Sachsen liegt, dann dauert die Logikvorlesung 90 Minuten.

2) Wenn Leipzig Hauptstadt von Bayern ist, dann ist Goethe mit Madonna verheiratet.

Beide Implikationen logisch gesehen wahr, weil dann-Teil von 1 wahr und wenn-Teil von 2 falsch. Beide Sätze würde man so nie äußern (Gricesche Konversationsmaximen, wenn ... dann drückt in der natürlichen Sprache häufig kausalen Zusammenhang aus, wird von Logik nicht erfasst)

2.2 Semantik der Aussagenlogik

Ziel: jede komplexe Formel in Abhängigkeit von den Wahrheitswerten der vorkommenden atomaren Formeln zu wahr oder falsch auswerten.

Als Wahrheitswerte verwenden wir 0 (falsch) und 1 (wahr).

Def.: (Belegung)

Eine (Wahrheits-) Belegung ist eine Funktion $I: A \rightarrow \{0,1\}$, wobei A die Menge der atomaren Formeln ist.

Bem.: Belegungen werden oft auch Interpretationen genannt.

Def.: (Wahrheitswert komplexer Formeln)

Sei I eine Belegung. Wir erweitern I zu einer Funktion $\hat{I}: E \rightarrow \{0,1\}$, wobei E die Menge aller Formeln ist, die aus atomaren Formeln in A aufgebaut sind:

1. Für jede atomare Formel $A_i \in A$ ist $\hat{I}(A_i) = I(A_i)$

2. $\hat{I}((F \wedge G)) = \begin{matrix} 1 \text{ falls } \hat{I}(F) = 1 \text{ und } \hat{I}(G) = 1 \\ 0 \text{ sonst} \end{matrix}$

3. $\hat{I}((F \vee G)) = \begin{matrix} 1 \text{ falls } \hat{I}(F) = 1 \text{ oder } \hat{I}(G) = 1 \\ 0 \text{ sonst} \end{matrix}$

4. $\hat{I}(\neg F) = \begin{matrix} 1 \text{ falls } \hat{I}(F) = 0 \\ 0 \text{ sonst} \end{matrix}$

Da \hat{I} Erweiterung von I ist, d.h. für atomare Formeln mit I übereinstimmt, schreiben wir vereinfachend I statt \hat{I} .

Beispiel:

Sei I eine Belegung, die A zu 1, B zu 0 und C zu 1 auswertet.

Wir bestimmen den Wahrheitswert der Formel $((A \vee B) \wedge (\neg C \vee \neg B))$.

$I(((A \vee B) \wedge (\neg C \vee \neg B))) = 1$ gdw

$I((A \vee B)) = 1$ und $I((\neg C \vee \neg B)) = 1$ gdw

$[I(A) = 1 \text{ oder } I(B) = 1]$ und $[I(\neg C) = 1 \text{ oder } I(\neg B) = 1]$ gdw

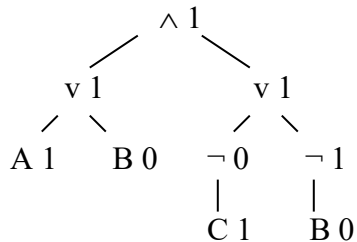
$[I(A) = 1 \text{ oder } I(B) = 1]$ und $[I(C) = 0 \text{ oder } I(B) = 0]$

Da $I(A) = 1$ und $I(B) = 0$ ist diese Bedingung erfüllt, d.h. die Formel wird zu wahr ausgewertet.

Weniger umständlich: wir ersetzen die atomaren Formeln durch ihre Wahrheitswerte und propagieren (zu beachten: die so entstehenden Konstrukte sind keine Formeln!):

$$\begin{aligned}
& ((1 \vee 0) \wedge (\neg 1 \vee \neg 0)) \\
& ((1 \vee 0) \wedge (0 \vee 1)) \\
& (1 \wedge 1) \\
& 1
\end{aligned}$$

entspricht folgender Baumdarstellung:



Wirkung der Junktoren lässt sich durch Wahrheitstafeln darstellen:

F	¬F
0	1
1	0

F	G	F ∧ G
0	0	0
0	1	0
1	0	0
1	1	1

F	G	F ∨ G
0	0	0
0	1	1
1	0	1
1	1	1

F	G	F → G
0	0	1
0	1	1
1	0	0
1	1	1

F	G	F ↔ G
0	0	1
0	1	0
1	0	0
1	1	1

Def. (Modell, Gültigkeit, Erfüllbarkeit)

Sei F eine Formel. Eine Belegung, die F zu 1 auswertet, heißt Modell von F.

Falls I Modell von F schreiben wir: $I \models F$, falls nicht: $I \not\models F$.

Eine Formel F heißt erfüllbar, wenn F mindestens ein Modell besitzt.

Unerfüllbare Formeln heißen auch widersprüchlich (Kontradiktionen).

F heißt allgemeingültig (Tautologie), falls jede Belegung Modell für F ist.

Notation: $\models F$ für F ist Tautologie, $\not\models F$ für F ist nicht Tautologie.

Eine Belegung ist Modell einer Menge von Formeln M, wenn sie jedes Element von M zu 1 auswertet. M ist erfüllbar, wenn M mindestens ein Modell besitzt.

Satz: F ist Tautologie gdw. $\neg F$ unerfüllbar.

Bew.: F Tautologie gdw jede Belegung wertet F zu 1 aus gdw keine Belegung wertet $\neg F$ zu 1 aus gdw $\neg F$ hat kein Modell.

allg.gültig	erfüllbar nicht allg.gültig	unerfüllbar
G	F $\neg F$	$\neg G$

Def.: (logische Folgerung)

Seien F_1, \dots, F_k, G Formeln. G folgt logisch aus $\{F_1, \dots, F_k\}$ gdw jede zu F_1, \dots, F_k , und G passende Belegung, die Modell von $\{F_1, \dots, F_k\}$ ist, auch Modell von G ist.

Falls G logisch aus $\{F_1, \dots, F_k\}$ folgt, so schreiben wir $\{F_1, \dots, F_k\} \models G$.

Eine passende belegung ist eine, die alle in F_1, \dots, F_k vorkommenden atomaren Formeln auf 0 oder 1 abbildet.

Satz: Folgende Aussagen sind äquivalent:

1. G folgt logisch aus $\{F_1, \dots, F_k\}$.
2. $((\dots(F_1 \wedge F_2) \wedge \dots \wedge F_k) \rightarrow G)$ ist Tautologie.
3. $((\dots(F_1 \wedge F_2) \wedge \dots \wedge F_k) \wedge \neg G)$ ist unerfüllbar.

Bew.: Übung/Folien

Beispiel: B folgt logisch aus $\{B \vee \neg C, C\}$

Wahrheitswert einer Formel F hängt nur von in F vorkommenden atomaren Formeln ab.

Wenn F n atomare Formeln enthält, so gibt es 2^n zu testende Belegungen. Diese können in Wahrheitstabeln systematisch beschrieben werden.

A	B	$(\neg A \vee B)$	$(\neg B \vee A)$	$(\neg A \vee B) \wedge (\neg B \vee A)$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	1	1	1

Rechte Spalte: Wahrheitsverlauf der jeweiligen Formel

Wahrheitstafelmethode im Allgemeinen sehr aufwändig: exponentiell viele Zeilen.

Wie viele Formeln mit n atomaren Formeln und verschiedenen Wahrheitsverläufen gibt es?

Tafel hat 2^n Zeilen, an jeder Stelle im Wahrheitsverlauf 1 oder 0, also: $2 \cdot 2^n$

Satz (Monotonie der Folgerungsrelation): Seien H und H' Mengen von Formeln, so dass H Teilmenge von H' . Falls F logisch aus H folgt, so auch aus H' .

Satz: Sei H eine unerfüllbare Menge von Formeln, F eine Formel. Es gilt $H \models F$.

2.3 Äquivalenz und Normalformen

Def.: (Äquivalenz)

Zwei Formeln F und G heißen äquivalent (Notation: $F \equiv G$), falls für alle Belegungen I gilt: $I(F) = I(G)$.

Satz: Seien F und G Formeln. $F \equiv G$ gdw. $F \leftrightarrow G$ ist Tautologie

Satz: (Ersetzbarkeit)

Seien F und G äquivalente Formeln. Sei H eine Formel, die F als Teilformel besitzt. H' entstehe aus H durch Ersetzen eines Vorkommens von F in H durch G . Dann gilt: $H \equiv H'$.

Beweis: Induktion über Formelaufbau von H .

Satz: Es gelten folgende Äquivalenzen:

$(F \wedge F) \equiv F$	
$(F \vee F) \equiv F$	(Idempotenz)
$(F \wedge G) \equiv (G \wedge F)$	
$(F \vee G) \equiv (G \vee F)$	(Kommutativität)
$((F \wedge G) \wedge H) \equiv (F \wedge (G \wedge H))$	
$((F \vee G) \vee H) \equiv (F \vee (G \vee H))$	(Assoziativität)
$(F \wedge (F \vee G)) \equiv F$	
$(F \vee (F \wedge G)) \equiv F$	(Absorption)
$(F \wedge (G \vee H)) \equiv ((F \wedge G) \vee (F \wedge H))$	
$(F \vee (G \wedge H)) \equiv ((F \vee G) \wedge (F \vee H))$	(Distributivität)
$\neg\neg F \equiv F$	(doppelte Negation)
$\neg(F \wedge G) \equiv (\neg F \vee \neg G)$	
$\neg(F \vee G) \equiv (\neg F \wedge \neg G)$	(de Morgansche Regeln)
$(F \vee G) \equiv F$, falls F Tautologie	
$(F \wedge G) \equiv G$, falls F Tautologie	(Tautologieregeln)
$(F \vee G) \equiv G$, falls F unerfüllbar	
$(F \wedge G) \equiv F$, falls F unerfüllbar	(Unerfüllbarkeitsregeln)

Beweis: Wahrheitstabeln (Beispiel nach Wahl)

Bemerkung 1: wegen Assoziativität können Klammern wegfallen :

$(A \vee B \vee C \vee D)$ anstelle von entsprechenden vollständig geklammerten Formeln.

Weitere Klammerersparnis durch Weglassen äußerer Klammern und Bindungskonventionen:

\neg stärker als \wedge stärker als \vee stärker als \rightarrow stärker als \leftrightarrow

Damit können wir schreiben $A \wedge B \vee C \rightarrow D$ statt $((A \wedge B) \vee C) \rightarrow D$

Bemerkung 2: Mit Hilfe der de Morganschen Regeln läßt sich jede Formel in eine äquivalente Formel ohne \vee bzw. ohne \wedge umwandeln. $\{\neg, \wedge\}$ bzw. $\{\neg, \vee\}$ genügen also eigentlich als Basis zum Aufbau der Syntax der Aussagenlogik (ebenso: $\{\neg, \rightarrow\}$ und andere Mengen).

Beispiel zum Gebrauch der Äquivalenzen:

$$\begin{aligned} \neg \text{Bier} \rightarrow \text{Fisch} & \qquad \text{Bier} \vee \text{Fisch} & (1) \\ \text{Eis} \vee \neg \text{Bier} \rightarrow \neg \text{Fisch} & \qquad \neg(\text{Eis} \vee \neg \text{Bier}) \vee \neg \text{Fisch} & (2) \end{aligned}$$

$$\begin{aligned} & (\text{Bier} \vee \text{Fisch}) \wedge (\neg(\text{Eis} \vee \neg \text{Bier}) \vee \neg \text{Fisch}) & \text{de Morgan} \\ \equiv & (\text{Bier} \vee \text{Fisch}) \wedge ((\neg \text{Eis} \wedge \neg \neg \text{Bier}) \vee \neg \text{Fisch}) & \text{doppelte Negation} \\ \equiv & (\text{Bier} \vee \text{Fisch}) \wedge ((\neg \text{Eis} \wedge \text{Bier}) \vee \neg \text{Fisch}) & \text{Kommutativität} \\ \equiv & (\text{Bier} \vee \text{Fisch}) \wedge (\neg \text{Fisch} \vee (\neg \text{Eis} \wedge \text{Bier})) & \text{Distributivität} \\ \equiv & (\text{Bier} \vee \text{Fisch}) \wedge ((\neg \text{Fisch} \vee \neg \text{Eis}) \wedge (\neg \text{Fisch} \vee \text{Bier})) & \text{Kommutativität} \\ \equiv & (\text{Bier} \vee \text{Fisch}) \wedge ((\text{Bier} \vee \neg \text{Fisch}) \wedge (\neg \text{Fisch} \vee \neg \text{Eis})) & \text{Assoziativität} \\ \equiv & ((\text{Bier} \vee \text{Fisch}) \wedge (\text{Bier} \vee \neg \text{Fisch})) \wedge (\neg \text{Fisch} \vee \neg \text{Eis}) & \text{Distributivität} \\ \equiv & (\text{Bier} \wedge (\text{Fisch} \vee \neg \text{Fisch})) \wedge (\neg \text{Fisch} \vee \neg \text{Eis}) & \text{Kommutativität} \\ \equiv & ((\text{Fisch} \vee \neg \text{Fisch}) \wedge \text{Bier}) \wedge (\neg \text{Fisch} \vee \neg \text{Eis}) & \text{Tautologieregel} \\ \equiv & \text{Bier} \wedge (\neg \text{Fisch} \vee \neg \text{Eis}) \end{aligned}$$

Def.: (Normalformen)

Ein Literal ist eine atomare Formel (positives Literal) oder die Negation einer atomaren Formel (negatives Literal).

Eine Formel F ist in konjunktiver Normalform (KNF), falls sie eine Konjunktion von Disjunktionen von Literalen ist.

Eine Formel F ist in disjunktiver Normalform (DNF), falls sie eine Disjunktion von Konjunktionen von Literalen ist.

Bemerkung 1: Konjunktionen und Disjunktionen in der obigen Definition können eine beliebige (endliche) Anzahl von Elementen enthalten. Das schließt auch Konjunktionen und Disjunktionen aus einer einzigen Teilformel mit ein (in diesem Fall kommt \wedge bzw. \vee gar nicht vor!).

Z.B. $A \wedge B$ ist sowohl in KNF (Konjunktion von 2 Disjunktionen aus jeweils einem Element) als auch in DNF (eine einstellige Disjunktion, die als Element die Konjunktion von A und B enthält).

Bemerkung 2: es gibt unterschiedliche Formeln in DNF (KNF), die äquivalent sind:

Beispiel (DNF)

$$\begin{aligned} (A \wedge B \wedge C) \vee (A \wedge B \wedge \neg C) \vee F & \text{ äquivalent zu} \\ (A \wedge B) \vee F & \end{aligned}$$

Satz: Für jede Formel F gibt es eine äquivalente Formel in KNF und eine äquivalente Formel in DNF.

Beweis durch Induktion über Formelaufbau von F.

Algorithmus zur Umformung einer Formel in KNF:

Gegeben: Formel F (Abkürzungen \rightarrow und \leftrightarrow seien bereits ersetzt)

1. Ersetze in F jedes Vorkommen einer Teilformel der Bauart

$$\begin{array}{ll} \neg\neg G & \text{durch } G \\ \neg(G \vee H) & \text{durch } (\neg G \wedge \neg H) \\ \neg(G \wedge H) & \text{durch } (\neg G \vee \neg H) \end{array}$$

bis keine solche Teilformel mehr vorkommt.

2. Ersetze in F jedes Vorkommen einer Teilformel der Bauart

$$\begin{array}{ll} (F \vee (G \wedge H)) & \text{durch } ((F \vee G) \wedge (F \vee H)) \\ ((F \wedge G) \vee H) & \text{durch } ((F \vee H) \wedge (G \vee H)) \end{array}$$

bis keine solche Teilformel mehr vorkommt (ausdistribuierten).

(Bei Umformung in DNF muss im letzten Schritt \vee und \wedge vertauscht werden)

In Schritt 1 wird die Negation schrittweise nach innen verschoben, bis sie nur noch vor Atomen vorkommt. In Schritt 2 wird die Disjunktion nach innen geschoben, bis die gewünschte Form erreicht ist.

Beispiel:

$(A \rightarrow B) \Rightarrow (B \Rightarrow C)$	Eliminieren \rightarrow
$\neg(A \Rightarrow B) \vee (\neg B \vee C)$	Eliminieren \rightarrow
$\neg(\neg A \vee B) \vee (\neg B \vee C)$	de Morgan, doppelte Negation
$(A \wedge \neg B) \vee \neg B \vee C$	schon in DNF, Distributivität
$((A \vee \neg B) \wedge (\neg B \vee \neg B)) \vee C$	Distributivität
$(A \vee \neg B \vee C) \wedge (\neg B \vee \neg B \vee C)$	KNF

Weitere Vereinfachung: wegen Idempotenz und da $A \wedge (A \vee B)$ äquivalent zu A :
 $(\neg B \vee C)$ (diese Schritte sind im obigen Algorithmus noch nicht enthalten)

Bemerkung: KNF und DNF können auch direkt aus Wahrheitstafel abgelesen werden.

DNF: Die Konjunktionen von Literalen entstehen aus Zeilen, für die der Wert von F 1 ist. Atome, deren Wert in der Zeile 1 ist, werden positive Literale, die anderen negative.

KNF: Die Disjunktionen von Literalen entstehen aus Zeilen, für die Wert von F 0 ist. Atome, deren Wert in der Zeile 0 ist, werden positive Literale, die anderen negative.

(dass das so korrekt ist sieht man wie folgt: Bilde DNF für $\neg F$, wende de Morgan an. Das liefert Formel $\neg G$, so dass G in KNF, also G KNF von F . In G wurden alle positiven Literale der DNF zu negativen und umgekehrt)

Beispiel:

ABC	$(A \rightarrow B)$	$(B \rightarrow C)$	$(A \rightarrow B) \rightarrow (B \rightarrow C)$
0 0 0	1	1	1
0 0 1	1	1	1
0 1 0	1	0	0
0 1 1	1	1	1
1 0 0	0	1	1
1 0 1	0	1	1
1 1 0	1	0	0
1 1 1	1	1	1

Ablesen DNF:

$$(\neg A \wedge \neg B \wedge \neg C) \vee (\neg A \wedge \neg B \wedge C) \vee \dots \vee (A \wedge B \wedge C) \quad \text{insgesamt 6 Konjunktionen}$$

Ablesen KNF: $(A \vee \neg B \vee C) \wedge (\neg A \vee \neg B \vee C)$
 äquivalent zu $(\neg B \vee C)$

Warum funktioniert das? DNF für $\neg F$:

$$D = (\neg A \wedge B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$$

Durch Negieren entsteht:

$$\begin{aligned} K &= \neg((\neg A \wedge B \wedge \neg C) \vee (A \wedge B \wedge \neg C)) \\ &\equiv \neg(\neg A \wedge B \wedge \neg C) \wedge \neg(A \wedge B \wedge \neg C) \\ &\equiv (A \vee \neg B \vee C) \wedge (\neg A \vee \neg B \vee C) \end{aligned}$$