051013 VO Theoretische Informatik

Einführung, Formale Logik in der Informatik, Aussagenlogik

Ekaterina Fokina

universität wien

Section 1

VO Theoretische Informatik - Ausblick

Abstraktion

Abstraktion ist ein zentrales Konzept in der Informatik:

Abstraktion

- Reduktion der vorhanden Information auf die für das aktuelle Problem wesentliche Information.
- oft auch: die Reduktion auf ein allgemeineres bzw. einfacheres Konzept.

Beispiel

Statt eine Landkarte zu betrachten verwendet man eine Graphen mit den Distanzen zwischen den relevanten Zielen.

Modell

- Vereinfachte abstrakte Darstellung eines komplexen Systems (eines Ausschnitts der „Realität“)
- um dieses einfacher zu verstehen und zu analysieren.
- Es werden nur die für den Zweck relevanten Aspekte darstellt.

Analyse von Systemen

- Modell kommt nach der Realität.
- Modell wird angepasst, bis es die Realität wiedergibt.
- Kritischer Vergleich der Voraussagen des Modells mit der Realität.

Planung/Spezifikation von Systemen

- Modell kommt vor dem System.
- System wird angepasst und korrigiert, bis es dem Modell entspricht.
- Kritischer Vergleich des Systemverhaltens mit dem Modell.

Natürliche Sprachen

Natürliche Sprachen sind

- universell,
- vielseitig,
- ausdrucksstark, und
- wandlungsfähig,

aber auch

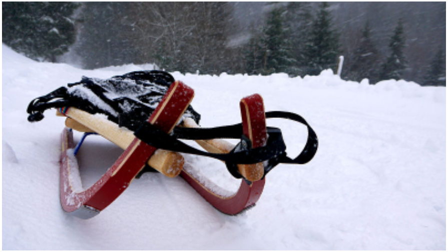
- komplex,
- mehrdeutig, und
- ungenau.

und daher nicht gut geeignet zur Modellierung und Spezifikation.

Natürliche Sprachen (Mehrdeutigkeit)

Mehrdeutigkeit einzelner Wörter: Rodel.

Rodelschlitten[a]



vs.

Sackrodel[b]

anderes Beispiel: Schimmel (Pilz oder Pferd?)

[a] Bild Von Basti007 - Eigenes Werk, CC BY 3.0, https://commons.wikimedia.org/w/index.php?curid=20206926

[b] Bild Gemeinfrei, https://commons.wikimedia.org/w/index.php?curid=971442.

Natürliche Sprachen (ungenau)

Beispiel

Der Satz: “Jede Frau liebt einen Mann” kann unterschiedlich interpretiert werden

- Jede Frau liebt mindestens einen Mann.
- Jede Frau liebt genau einen Mann.
- Alle Frauen lieben den selben Mann.

Formale Sprachen

Kernaspekte formaler Sprachen:

- Syntax: Welche Äußerungen sind in der Sprache gültig?
- Semantik: Was bedeuten diese gültigen Äußerungen?
- Ausdrucksstärke: Was kann in der Sprache ausgedrückt werden?

Formale Sprachen in der THI-Vorlesung

- Logische Sprachen (Teil 1)
  - Aussagenlogik
  - Prädikatenlogik
  - Prolog (Logische Programmierung)
- Reguläre Ausdrücke (Teil 2)
- Formale Grammatiken (Teil 2)
- Automaten & Turing Maschinen (Teil 2)

Heute

- Logik in der Informatik: ein Überblick
- Aussagenlogik:
    - Syntax
    - Semantik
    - Grundbegriffe
    - Eigenschaften
    - Formalisieren in Aussagenlogik: Beispiele

Section 2

Formale Logik in der Informatik

Formale Logik in der Informatik

Das Gebiet der Logik

- beschäftigt sich mit den Prinzipien des korrekten Schließens.
- hat Überschneidungen mit vielen Wissenschaftsdisziplinen: Philosophie, Mathematik, Rechtswissenschaften, ..., und Informatik.

Wir interessieren uns für formale Logik (auch mathematische Logik)

- Basiert auf Formalen Sprachen.
- Studiert die Gültigkeit von Ableitungs- und Folgerungsbeziehungen.

Ergänzende/Weiterführende Literatur

Die Vorlesung folgt dem Buch von Kreuzer & Kühling

Martin Kreuzer, Stefan Kühling (2006).
Logik für Informatiker.
Addison-Wesley Verlag, ISBN-13: 978-3827372154

Freie Ressourcen:

Wikibooks
Logic for Computer Science.
http://en.wikibooks.org/wiki/Logic_for_Computer_Science

In der Fachbereichsbibliothek:

Jürgen Dassow (2005).
Logik für Informatiker.
Vieweg+Teubner Verlag, ISBN-13: 978-3519005186

Hartmut Ehrig et al. (2001).
Mathematisch-strukturelle Grundlagen der Informatik. (Teil III & IV)
Springer Verlag, ISBN-13: 978-3540419235

Worum geht es in der Logik?

Es geht um

- das Ziehen von **Schlussfolgerungen**
- die Gültigkeit von **Begründungen**
- die **Widerspruchsfreiheit** zwischen Aussagen

Die formale Logik befasst sich **nicht**

mit dem Wahrheitsgehalt/Inhalt einer Aussage

an sich, sondern mit Regeln die es erlauben

aus wahren Aussagen richtige Schlüsse zu ziehen.

Worum geht es in der Logik?

Beispiel

Aussage 1: Jeder Mensch hat 2 Augen.
Aussage 2: John ist ein Mensch.
Schlussfolgerung: Also hat John 2 Augen.

Beispiel

Aussage 1: Jeder Mensch hat 4 Augen.
Aussage 2: John ist ein Mensch.
Schlussfolgerung: Also hat John 4 Augen.

Beobachtungen:

1. Wir haben die gleiche (gültige) Regel zum Schlussfolgern verwendet.
2. Aber bei falschen Prämissen haben wir als Schluss eine falsche Aussage bekommen, obwohl die Regel korrekt ist.

Bemerkung

Im Allgemeinen kann es auch passieren, dass der Schluss zufälligerweise korrekt ist, auch wenn die Prämissen falsch sind. Dazu später in der Vorlesung.

Worum geht es in der Logik?

Die formale Logik beschäftigt sich damit unter welchen Bedingungen man aus der

Gültigkeit von Voraussetzungen

auf die

Gültigkeit von Folgerungen

schließen kann.

Dazu nutzen wir in der Informatik unterschiedliche logische Systeme (oft auch als unterschiedliche "Logiken" bezeichnet).

Logische Systeme

Bestandteile eines logischen Systems:

1. **Syntax**: Legt fest, welche formalen Ausdrücke als Formeln des logischen Systems gelten. Üblicherweise startet man von sogenannten atomaren Formeln und legt Regeln fest, wie aus diesen weitere Formeln zusammengesetzt werden dürfen.
2. **Semantik**: Regeln, die festlegen, wie Formeln Wahrheitswerte zugeordnet werden können. Dem syntaktischen Element einer Formel wird damit eine Bedeutung gegeben.
3. **Kalkül**: Ein System von Regeln zum (systematischen) Ableitung neuer wahrer Formeln.

Logische Systeme, Beispiele

In der Vorlesung werden wir zwei logische Systeme kennen lernen

- Aussagenlogik
- Prädikatenlogik

Es gibt aber noch unzählige andere logische Systeme (mit Bedeutung in der Informatik)

- Mehrwertige Logiken und Fuzzylogik
- Nichtmonotone Logiken (z.B.: Reitersche Default-Logik)
- Modallogik
- ...

Anwendungen in der Informatik

- Boolesche Ausdrücke in Programmiersprachen (if Anweisungen)
- Formale Spezifikation
- Verifikation von Programmen
- Logische Programmierung
- Datenbanksysteme
- Wissensrepräsentation und Künstliche Intelligenz
- Berechenbarkeits- und Komplexitätstheorie
- ...

Section 3

Aussagenlogik

Aussagenlogik - Einleitung

Eine **Aussage** ist

- Ein Satz in der natürlichen Sprache (z.B. “Das Auto ist rot.”)
- Ist entweder wahr (W) oder falsch (F)

Beispiel

„4 ist größer als 10“	F
„101 ist eine Primzahl“	W
„Junge Pferde nennt man Welpen“	F
„4 ist größer als 10, oder 10 ist größer als 4“	W
„4 ist größer als 10, und 10 ist größer als 4“	F
„Die Vorlesung ist im AudiMax“	?
„Es gibt unendlich viele Primzahlzwillinge“	?

Aussagenlogik - Einleitung

Es gibt Aussagen, die aus der logischen Verknüpfung von mehreren einfacheren Aussagen bestehen. Wir wollen diese mit atomaren Aussagen und logischen Operatoren darstellen.

Eine **einfache/atomare Aussage**

- ist eine Aussage
- die nur einen Sachverhalt enthält (unteilbare Aussage).

Insbesondere enthält sie keine aussagenlogischen Verknüpfungen wie: nicht, und, oder, wenn ... dann, genau dann wenn

Als **logische Operatoren / Verknüpfungen** betrachten wir

- die Negation $\neg A$ (NOT),
- die Konjunktion $A \wedge B$ (AND),
- die Disjunktion $A \vee B$ (OR), und
- die Folgerung/Implikation $A \rightarrow B$.

Aussagenlogik - Einleitung

Beispiel

Die Aussage „Das Auto ist rot und das Fahrrad ist grün“ besteht aus zwei atomare Aussagen:

1. „Das Auto ist rot“
2. „Das Fahrrad ist grün“

Mit einer Konjunktion können wir die ursprüngliche Aussage aus den Atomen aufbauen.

$$\text{„Das Auto ist rot“} \land \text{„Das Fahrrad ist grün“}$$

Beispiel

Aussage: „Gestern hat es in Wien geregnet“

Die Aussage ist atomar. Sie enthält zwar mehrere Informationen (Zeit, Ort, Wetter), beschreibt aber nur einen Sachverhalt und kann nicht in mehrere Aussagen aufgespalten werden.

Aussagenlogik - Einleitung

Beispiele:

Beispiel

„4 ist größer als 10, oder 10 ist größer als 4“
„4 ist größer als 10” $\vee$ “10 ist größer als 4“

Beispiel

„Wenn es regnet ist die Straße nass.“
„Es regnet” $\rightarrow$ “Die Straße ist nass.“

Beispiel

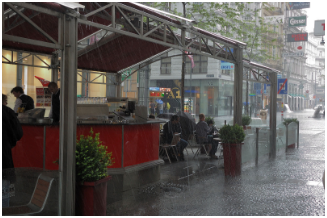
„Wenn ich nicht lerne, schaffe ich die Prüfung nicht.“
($\neg$ “Ich lerne.”) $\rightarrow$ ($\neg$ „Ich schaffe die Prüfung.“)
(„Ich lerne.”) $\vee$ ($\neg$ „Ich schaffe die Prüfung.“)

Implikation

- Grundfrage: Wie hängen Aussagen zusammen?
- Implikation $A \to B$ („A impliziert B", „wenn A dann B", „A hinreichend für B", „B notwendig für A")
- Beispiele der Form $A \to B$:
  1. $x = 1 \to x \cdot y = y$.
  2. a, b sind Katheten und c Hypothenuse in einem rechtwinkeligen Dreieck $\to a^2 + b^2 = c^2$.
  3. Aber auch: $x = 0 \to a + b = b + a$
- Hier: A, B sind jeweils nicht immer wahr, aber $A \to B$ ist immer wahr.
- Genauer: $A \to B$ ist falsch falls: A wahr ist und B falsch ist.
- Sonst ist $A \to B$ immer wahr!

Beispiel

Wenn es regnet, wird die Straße nass.



Äquivalenz

- Grundfrage: Wie hängen Aussagen zusammen?
- Äquivalenz $A \leftrightarrow B$ („A genau dann, wenn B")
- $A \leftrightarrow B$ definiert als: $A \rightarrow B$ und $B \rightarrow A$ gelten gleichzeitig.
- Beispiele der Form $A \leftrightarrow B$:
  1. $x = 1 \leftrightarrow x^2 = 1$: Gilt nicht, da zwar $A \rightarrow B$ aber $B \not\rightarrow A$.
  2. $x + y = 1 \leftrightarrow y + x = 1 + 2 \cdot 0$: Gilt (denn $1 = 1 + 2 \cdot 0$).

Aussagen

- Die ,,wichtigsten" Aussagen in der Logik sind Äquivalenzen und Implikationen.
- Äquivalenz zwischen einer komplizierten und einer einfachen Aussage:

$$x^2 - 5x + 6 = 0 \leftrightarrow (x = 2 \vee x = 3).$$

- Implikation drückt ,,Wissen über Spezialfälle" aus:

Theorem (Nullstellensatz v. Bolzano)

Sei f eine Funktion stetig auf $[a, b]$ und $f(a) \leq 0 \leq f(b)$. Dann gibt es ein y sodass $a \leq y \leq b$ und $f(y) = 0$.

Beispiel

Sei A eine Aussage "n ist ein Vielfaches von 10". Welche von den folgenden Aussagen sind notwendig für A? Hinreichend für A? Äquivalent zu A?

B. n ist ein Vielfaches von 5;

C. n ist durch 20 teilbar;

D. n ist gerade und durch 5 teilbar;

E. $n = 100$

F. n^2 ist durch 100 teilbar.

Aussagen

- Grundfrage: Wie hängen Aussagen zusammen?
- Wie findet man gültige Zusammenhänge?
- Mittels Beweis!
- Beweis ist eine Kette von wahren Implikationen.

Beweise

- Jeder Beweisschritt soll offensichtlich sein:
- Entweder logisch evident, oder
- schon vorher bewiesen bzw. bekannt.

Beweise

- Man unterscheidet 2 Arten von Beweisen von $A \to B$:
- *Direkter Beweis.* Unter der Annahme A wird durch logisches Schließen B hergeleitet.
- *Indirekter Beweis (Widerspruchsmethode).* Unter Annahme von A und $\neg B$ wird ein Widerspruch (z.B. $0 = 1$) hergeleitet.

Beweise

- *Direkter Beweis.* Unter der Annahme A wird durch logisches Schließen B hergeleitet.
- Beispiel:

Sei x, y rationale Zahlen. Z.z.: $x + y$ ist auch rational.

Beweis.

- Wir nehmen an, dass die Voraussetzung wahr ist: x, y sind rational.
- Wenn x rational ist, dann $x = \frac{m}{p}$, wo m, p ganze Zahlen sind.
- Wenn y rational ist, dann $y = \frac{n}{q}$, wo n, q ganze Zahlen sind.
- Dann $x + y = \frac{mq+np}{pq}$.
- Deshalb ist $x + y$ auch rational. Die Behauptung ist also wahr.
- Die ganze Implikation ist also wahr.

$\square$

Beweise

- *Indirekter Beweis.* Unter Annahme von A und $\neg B$ wird ein Widerspruch (z.B. $0 = 1$) hergeleitet.
- Beruht auf Wahrheit von $((A \wedge \neg B) \to 0 = 1) \to (A \to B)$ (Mittels Wahrheitstabelle nachrechnen!)
- Beispiel: Wir wollen beweisen, dass die Summe von drei aufeinanderfolgenden natürlichen Zahlen immer durch 3 teilbar ist.
- A=" n ist eine natürliche Zahl".
  B= " $n + (n + 1) + (n + 2)$ ist durch 3 teilbar".
- Wir nehmen an, dass $A \wedge \neg B$ wahr ist: n ist eine natürliche Zahl und $n + (n + 1) + (n + 2)$ ist durch 3 nicht teilbar.
- $n + (n + 1) + (n + 2) = 3n + 3 = 3(n + 1)$.
- Also, $3(n + 1)$ ist durch 3 nicht teilbar.
- Dann ist $n + 1$ keine natürliche Zahl.
- Dann ist auch n keine natürliche Zahl. Widerspruch!

Aussagenlogik - Formale Logik

Zur Erinnerung: In der formalen Logik müssen wir alle Bestandteile eines logischen Systems genau definieren.

Auf den folgenden Folien:

- **Formale Syntax**: Wir definieren, was genau eine Aussagenlogische Formel ist.
- **Formale Semantik**: Wir geben Aussagenlogischen Formeln eine Bedeutung.

Später:

- **Kalkül & Schlussregeln**

Subsection 1

Die Syntax der Aussagenlogik

Induktive Definition

- $A \ldots$ Menge von Atomen /Grundelementen
- Funktionen $f(x, y)$, $g(x_1, \ldots, x_n)$, $\ldots$

Induktive Definition einer Menge / Sprache L

L ist die kleinste Menge für die folgendes gilt:

- $A \subseteq L$
- Wenn $x, y \in L$ dann ist auch $f(x, y) \in L$
- Wenn $x_1, \ldots, x_n \in L$ dann ist auch $g(x_1, \ldots, x_n) \in L$

Beispiel

Die Menge $\mathbb{G}$ der geraden Zahlen ist induktiv wie folgt definiert.

- $0 \in \mathbb{G}$
- Wenn $x \in \mathbb{G}$ dann ist auch $x + 2 \in \mathbb{G}$

Wichtig: $\mathbb{G}$ ist die kleinste Menge, die beide Bedingungen erfüllt.

Aussagenlogische Formeln - Syntax

Nun wollen wir formal definieren, was wir unter Aussagenlogischen Formeln (auch Boolesche Ausdrücke, nach George Boole) verstehen.

Definition (Aussagenlogische Formel)

Wir betrachten eine Menge von **atomaren Formeln** (Variablen) Var. Die aussagenlogischen Formeln sind jetzt induktiv definiert.

- Jede atomare Formel aus Var ist eine Formel
- Sind F und G Formeln dann sind auch

  $$\neg F, \quad (F \wedge G), \quad (F \vee G), \quad \text{und} \quad (F \rightarrow G)$$

  Formeln.

$\mathcal{F}_{\mathit{Var}}$ bezeichnet die Menge der aussagenlogischen Formeln, die aus den Atomen Var gebildet werden können.

Aussagenlogische Formeln - Syntax

Beispiel

Für die atomaren Formeln $Var = \{x, y, z\}$ können wir z.B. die folgende Formel bilden:

$$((x \to y) \land (z \lor \neg x))$$

Zunächst können wir Formeln $F_1 = x$ und $F_2 = y$ bilden und dann

$$F_3 = (F_1 \to F_2) = (x \to y)$$

Mit $F_4 = z$ und $F_5 = \neg F_1 = \neg x$ bilden wir dann

$$F_6 = (F_4 \lor F_5) = (z \lor \neg x)$$

Schlussendlich erhalten wir

$$(F_3 \land F_6) = ((x \to y) \land (z \lor \neg x))$$

Subsection 2

Die Semantik der Aussagenlogik

Semantik der Aussagenlogik

Die Semantik der Aussagenlogik wird über Wahrheitswerte “wahr” (1) und “falsch” (0) definiert.

Die Aussagenlogik ist damit eine zweiwertige Logik [1].

Ein $x \in Var$ steht für irgendeine Aussage. Der Wahrheitswert von x hängt also davon ab für welche konkrete Aussage x steht.

- Der Wahrheitswert einer Formel wird in Abhängigkeit von den Wahrheitswerten der atomaren Formeln berechnet.
- Jeder Formel wird dann entweder der Wert 1 oder 0 zugeordnet.

Definition

Eine **Belegung** ist eine Funktion $\alpha : Var \to \{1, 0\}$ die

- jeder atomaren Formel entweder den Wert 1 oder 0 zuweist.

Diese Wahrheitswerte setzten sich auf beliebige Formeln fort.

---

[1] Außerhalb dieser Vorlesung gibt es sogenannte mehrwertige Logiken, die mehr als zwei Wahrheitswerte betrachten.

Semantik der Aussagenlogik

Definition

Sei α eine Belegung für Atome in Var. Wir definieren die Erweiterung $\widehat{\alpha}$ der Belegung α auf alle Formeln in $\mathcal{F}_{Var}$ wie folgt:

- für atomare Formeln $F \in Var$: $\widehat{\alpha}(F) = \alpha(F)$
- für Formeln $F, G \in \mathcal{F}_{Var}$:
  $$\widehat{\alpha}(F \wedge G) = \begin{cases} 1 & \text{Wenn } \widehat{\alpha}(F) = 1 \text{ und } \widehat{\alpha}(G) = 1 \\ 0 & \text{sonst} \end{cases}$$
  $$\widehat{\alpha}(F \vee G) = \begin{cases} 1 & \text{Wenn } \widehat{\alpha}(F) = 1 \text{ oder } \widehat{\alpha}(G) = 1 \\ 0 & \text{sonst} \end{cases}$$
  $$\widehat{\alpha}(\neg F) = \begin{cases} 1 & \text{Wenn } \widehat{\alpha}(F) = 0 \\ 0 & \text{sonst} \end{cases}$$
- für Formeln $F, G \in \mathcal{F}_{Var}$: $\widehat{\alpha}(F \rightarrow G) = \widehat{\alpha}(\neg F \vee G)$

Negation

Beispiel

Die Vorlesung ist nicht im AudiMax.

Die Aussage

- ist falsch wenn die Vorlesung im AudiMax ist
- und wahr andernfalls.

Negation

Der Wahrheitswert von $\neg F$ ergibt sich aus dem Wahrheitswert von F entsprechend der folgenden Wahrheitstafel:

$\hat{\alpha}(F)$	$\hat{\alpha}(\neg F)$
0	1
1	0

Andere Bezeichnungen/Symbole: not, non, $-F$, $\sim F$, $\bar{F}$, NF, ...

Konjunktion

Beispiel

Ich sehe mir einen Film an und esse Popcorn.

Ist nur wahr wenn beide Aussagen wahr sind.

Konjunktion

$\hat{\alpha}(F)$	$\hat{\alpha}(G)$	$\hat{\alpha}(F \wedge G)$
0	0	0
0	1	0
1	0	0
1	1	1

Andere Bezeichnungen/Symbole: and, et, $F \cdot G$, FG, $F\&G$, KFG, ...

Disjunktion

Beispiel

Ich sehe mir einen Film an oder esse Burger.

Ist wahr wenn mind. eine der beiden Aussagen wahr ist.

Disjunktion

$\hat{\alpha}(F)$	$\hat{\alpha}(G)$	$\hat{\alpha}(F \vee G)$
0	0	0
0	1	1
1	0	1
1	1	1

Andere Bezeichnungen/Symbole: or, vel, $F + G$, $F|G$, AFG, ...

Implikation

Beispiel

Wenn Fritz Fußball spielt, dann spielt auch Barbara Fußball.

Ist nur falsch wenn Fritz spielt aber Barbara nicht spielt

Implikation

$\hat{\alpha}(F)$	$\hat{\alpha}(G)$	$\hat{\alpha}(F \to G)$
0	0	1
0	1	1
1	0	0
1	1	1

Andere Bezeichnungen/Symbole: implies, only if, $F \supset G$, $F \Rightarrow G$, CFG, ...

Semantik der Aussagenlogik - Zusammenfassung

Für Formeln $F, G \in \mathcal{F}_{Var}$:

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \wedge G)$
0	0	0
0	1	0
1	0	0
1	1	1

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \vee G)$
0	0	0
0	1	1
1	0	1
1	1	1

$\widehat{\alpha}(F)$	$\widehat{\alpha}(\neg F)$
0	1
1	0

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \rightarrow G)$
0	0	1
0	1	1
1	0	0
1	1	1

Da alle Formeln nur aus solchen Verknüpfungen aufgebaut sind, ist damit $\widehat{\alpha}$ für alle Formeln definiert.

Semantik der Aussagenlogik - Äquivalenz Operator

Wir können jetzt weitere zusätzliche Operator definieren.

Beispiel

Fritz spielt Fußball genau dann wenn Barbara auch Fußball spielt.

Ist wahr wenn beiden spielen oder keiner der beiden spielt.

Äquivalenz

$(F \leftrightarrow G)$

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \leftrightarrow G)$
0	0	1
0	1	0
1	0	0
1	1	1

$F \leftrightarrow G$ ist also eine Kurzschreibweise von $(F \rightarrow G) \wedge (G \rightarrow F)$.
Andere Bezeichnungen/Symbole: iff, eq, $F \equiv G$, $F \Leftrightarrow G$, EFG, ...

Ausschließende Disjunktion (XOR)

Beispiel

Ich sehe mir entweder einen Film an oder spiele Fußball
(aber nicht beides).

Ist wahr wenn genau eine der beiden Aussagen wahr ist.

Ausschließende Disjunktion

Wir notieren die Ausschließende Disjunktion von zwei Aussagen F, G als $F \oplus G$.

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \oplus G)$
0	0	0
0	1	1
1	0	1
1	1	0

Andere Bezeichnungen/Symbole: Antivalenz, xor, $F \not\equiv G$, $F \not\Leftrightarrow G$,
$F \not\leftrightarrow G$, JFG, ...

Implikation (Umkehrung)

Beispiel

Fritz spielt Fußball, wenn Barbara Fußball spielt.

Ist nur falsch wenn Fritz nicht spielt aber Barbara spielt

Implikation

Wir notieren die Implikation von zwei Aussagen F, G als $F \leftarrow G$.

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \leftarrow G)$
0	0	1
0	1	0
1	0	1
1	1	1

Andere Bezeichnungen/Symbole: if, $F \subset G$, $F \Leftarrow G$, ...

NAND & NOR

Beispiel

- Es ist nicht der Fall, dass das Kuvert signiert und zugeklebt ist.
- Weder Fritz noch Barbara spielen Tischtennis.

Negierte Konjunktion & Negierte Disjunktion

Negierte Konjunktion: F nand G
Negierte Disjunktion: F nor G

$\widehat{\alpha}(F)$	$\widehat{\alpha}(G)$	$\widehat{\alpha}(F \wedge G)$	$\widehat{\alpha}(F \text{ nand } G)$	$\widehat{\alpha}(F \vee G)$	$\widehat{\alpha}(F \text{ nor } G)$
0	0	0	1	0	1
0	1	0	1	1	0
1	0	0	1	1	0
1	1	1	0	1	0

Andere Bezeichnungen/Symbole:

- NAND: Sheffer-Strich, $F \uparrow G$, $F|G$, F/G, DFG, ...
- NOR: Peirce-Pfeil, $F \downarrow G$, XFG, ...

Logische Operatoren – Zusammenfassung

Logische / Boolesche Operatoren

Konstanten

$\top$	$\bot$
1	0

Unäre Op.

x	$\neg x$
0	1
1	0

Binäre Operatoren

x	y	$x \wedge y$	x nand y	$x \vee y$	x nor y	$x \leftrightarrow y$	$x \oplus y$	$x \rightarrow y$	$x \leftarrow y$	$x \not\rightarrow y$	$x \not\leftarrow y$
0	0	0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	0	0	1	1	0	0	1
1	0	0	1	1	0	0	1	0	1	1	0
1	1	1	0	1	0	1	0	1	1	0	0

- Es gibt 2^{2^n} verschiedene n-stellige Boolesche Operatoren.
  - Es gibt 2^n Möglichkeiten für die Argumente
  - Jeweils 2 Möglichkeiten (0,1) für das Ergebnis

Funktionale Vollständigkeit

Definition (Funktionale Vollständigkeit)

Wie nennen eine Teilmenge der Booleschen Funktionen *funktional vollständig*, wenn damit alle Booleschen Funktionen ausgedrückt werden können.

Satz

Die Menge $\{\neg, \wedge, \vee\}$ *ist funktional vollständig.*

z.B.: $x \to y = \neg x \vee y$

Satz

Die Menge $\{\neg, \wedge\}$ *ist funktional vollständig.*

Funktionale Vollständigkeit

Satz

Die Menge $\{\neg, \wedge\}$ *ist funktional vollständig.*

Beweis:

- Wir wissen, dass die Menge $\{\neg, \wedge, \vee\}$ vollständig ist
- Es genügt also zu zeigen, dass $\vee$ mit Hilfe von $\neg$ und $\wedge$ ausgedrückt werden kann
- $x \;\vee\; y = \neg(\neg x \wedge \neg y)$ – das können wir mit einer Wahrheitstafel überprüfen

x	y	$\neg x$	$\neg y$	$\neg x \wedge \neg y$	$\neg(\neg x \wedge \neg y)$
0	0	1	1	1	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	0	1

Semantik der Aussagenlogik - Modelle

Definition

Eine Belegung α für *Var* ist ein **Modell einer Formel** F über *Var*, wenn $\widehat{\alpha}(F) = 1$, wir schreiben auch $\alpha \models F$.

Die Semantik einer Formel ergibt sich aus der Menge ihrer Modelle.

Definition

Eine Belegung α für *Var* ist ein **Modell einer Menge $\mathcal{F}$ von Formeln** wenn $\widehat{\alpha}(G) = 1$ für jede Formel $G \in \mathcal{F}$, wir schreiben auch $\alpha \models \mathcal{F}$.

Subsection 3

Aussagenlogik: Grundbegriffe

Definition

Eine Formel F ist **erfüllbar** wenn F mind. ein Modell hat (sonst **unerfüllbar**).

Definition

Eine Menge $\mathcal{F}$ von Formeln ist

- **konsistent** (**widerspruchsfrei**) wenn es ein Modell für $\mathcal{F}$ gibt.
- **inkonsistent**(**widersprüchlich**) wenn es kein Modell für $\mathcal{F}$ gibt.

Beispiel

Belegung α mit $\alpha(a) = 1$, $\alpha(b) = 1$, $\alpha(c) = 0$

- α ist ein Modell von $(a \vee c) \wedge (b \vee c)$
- Die Formel $(a \vee c) \wedge (b \vee c)$ ist daher erfüllbar
- Die Formelmenge $\{(a \vee c), (b \vee c)\}$ ist konsistent

Definition

Eine Formel F ist **allgemein gültig** / eine **Tautologie** wenn jede Belegung auch ein Modell ist.

Beispiele für Tautologien:

- $(x \vee \neg x)$
- $((a \vee c) \vee \neg c)$
- $(c \rightarrow c)$
- $(((a \rightarrow b) \wedge (b \rightarrow c)) \rightarrow (a \rightarrow c))$

Definition

Zwei Formeln F und G sind **semantisch äquivalent** wenn $\widehat{\alpha}(F) = \widehat{\alpha}(G)$ für alle Belegungen α. Wir schreiben $F \equiv G$.

Beispiele für semantisch äquivalente Formeln:

- $(x \wedge y)$ und $(y \wedge x)$ $\qquad (x \wedge y) \equiv (y \wedge x)$
- $(\neg x \vee y)$ und $(x \rightarrow y)$ $\qquad (\neg x \vee y) \equiv (x \rightarrow y)$
- $((a \vee b) \wedge (c \vee \neg c))$ und $(a \vee b)$ $\qquad ((a \vee b) \wedge (c \vee \neg c)) \equiv (a \vee b)$

Definition

Eine Formel G **folgt** aus der Formel F / der Formelmenge $\mathcal{F}$ wenn jedes Modell von F / $\mathcal{F}$ auch Modell von G ist.

- Wir schreiben $F \models G$ bzw. $\mathcal{F} \models G$.

Beispiele:

- Aus a folgt $(a \lor b)$. $\qquad a \models (a \lor b)$
- Aus $(\neg a \land b)$ folgt $(a \lor b)$. $\qquad (\neg a \land b) \models (a \lor b)$
- Aus $\{a, b\}$ folgt $(a \land b)$. $\qquad \{a, b\} \models (a \land b)$
- Aus $\{a, a \to b\}$ folgt b. $\qquad \{a, a \to b\} \models b$
- Aus $\{\neg b, a \to b\}$ folgt $\neg a$. $\qquad \{\neg b, a \to b\} \models \neg a$

G ist eine Tautologie $\iff \emptyset \models G$. In diesem Fall schreiben wir

$$\models G.$$

Grundlegende Begriffe - Zusammenfassung

α: Belegung für *Var*,
F,G: Formeln über *Var*
$\mathcal{F}$: Eine Menge von Formeln über *Var*

Definition

- $\mathcal{F}$ ist **konsistent** (**widerspruchsfrei**) wenn es ein Modell für $\mathcal{F}$ gibt.
  $\mathcal{F}$ ist **inkonsistent**(**widersprüchlich**) wenn es kein Modell für $\mathcal{F}$ gibt.
- F ist **erfüllbar** wenn F mind. ein Modell hat (sonst **unerfüllbar**).
- F ist **allgemein gültig** / eine **Tautologie** wenn jede Belegung auch ein Modell ist.
- F ist **semantisch äquivalent** zu G wenn $\widehat{\alpha}(F) = \widehat{\alpha}(G)$ für alle Belegungen α. Wir schreiben $F \equiv G$.
- G **folgt** aus $F/\mathcal{F}$ wenn jedes Modell von $F/\mathcal{F}$ auch Modell von G ist. Wir schreiben $F \models G$ bzw. $\mathcal{F} \models G$.

Formalisieren in Aussagenlogik

Unser ursprüngliches Ziel ist es (natürlich sprachliche) Aussagen zu formalisieren und logische Zusammenhänge zu identifizieren.

Faustregel zur Formalisierung von Deutsch:

Identifizieren von atomaren Aussagen:

- die kürzeste Aussage, in der keine logische Verknüpfung vorkommt (nicht, und, oder, wenn ... dann usw.), und
- die entweder wahr oder falsch sein kann

Dann ersetzt man atomare Aussagen mit Aussagenvariablen

Wörter/Phrasen nicht, und, oder, wenn ... dann, usw.

- werden durch die entsprechenden logischen Operationen ersetzt.

Formalisieren in Aussagenlogik - Beispiel 1

Aussage 1: Der Verkauf von Häusern sinkt, wenn die Zinsen steigen.
Aussage 2: Auktionäre sind nicht glücklich, wenn der Verkauf von Häusern sinkt.
Aussage 3: Die Zinsen steigen.
Aussage 4: Auktionäre sind glücklich.

Atomare Aussagen:
S: der Verkauf von Häusern sinkt
R: die Zinsen steigen
H: Auktionäre sind glücklich

Aussage 1: $R \to S$
Aussage 2: $S \to \neg H$
Aussage 3: R
Aussage 4: H

Wir wollen wissen, ob die Aussagen miteinander konsistent sind. Deshalb betrachten wir deren Konjunktion $(R \to S) \wedge (S \to \neg H) \wedge R \wedge H$.

Formalisieren in Aussagenlogik - Beispiel 1

Da wir wissen wollen ob die Aussagen miteinander konsistent sind betrachten wir deren Konjunktion $(R \rightarrow S) \wedge (S \rightarrow \neg H) \wedge R \wedge H$.

Wir testen ob die Formel erfüllbar ist und nutzen dazu Wahrheitstafeln.

S	R	H	$(R \rightarrow S)$	$(S \rightarrow \neg H)$	$(R \rightarrow S) \wedge (S \rightarrow \neg H) \wedge R \wedge H$
0	0	0	1	1	0
0	0	1	1	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	0	0
1	1	0	1	1	0
1	1	1	1	0	0

Die Formel hat kein Model (ist unerfüllbar) und daher sind die Aussagen widersprüchlich.

Formalisieren in Aussagenlogik - Beispiel 2

Aussage 1: Wenn der Geiger das Konzert gibt, werden viele kommen, wenn die Preise nicht zu hoch sind.
Aussage 2: Wenn der Geiger das Konzert gibt, werden die Preise nicht zu hoch sein.
Schluss: Daher werden, falls der Geiger das Konzert bestreitet, viele kommen.

Wir wollen wissen ob der Schluss korrekt ist.

Atomare Aussagen:
P: „der Geiger gibt das Konzert“
C: „viele werden kommen“
H: „die Preise sind zu hoch“

Aussage 1: $P \rightarrow (\neg H \rightarrow C)$
Aussage 2: $P \rightarrow \neg H$
Schluss: $P \rightarrow C$

Wir wollen wissen ob $P \rightarrow C$ aus $P \rightarrow (\neg H \rightarrow C)$ und $P \rightarrow \neg H$ folgt.

Indirekter Beweis/Widerspruchsmethode

Um $\{A, B\} \models C$ zu zeigen, können wir zeigen, dass

- $F = (A \wedge B) \rightarrow C$ eine Tautologie ist.

Die Formel F ist äquivalent zu

- $F' = \neg A \vee \neg B \vee C$.

Um zu testen ob F' eine Tautologie ist können wir testen ob $\neg F'$ unerfüllbar ist.

- $\neg F' \equiv A \wedge B \wedge \neg C$.

Wir erhalten also dass $\{A, B\} \models C$ genau dann gilt wenn $A \wedge B \wedge \neg C$ unerfüllbar ist.

Formalisieren in Aussagenlogik - Beispiel 2

Zurück zum Beispiel 2: Wir wollen wissen ob $P \rightarrow C$ aus $P \rightarrow (\neg H \rightarrow C)$ und $P \rightarrow \neg H$ folgt. (Wir schreiben auch $\{P \rightarrow (\neg H \rightarrow C), P \rightarrow \neg H\} \models P \rightarrow C$.)

Widerspruchsmethode:

Um $\{A, B\} \models C$ zu zeigen, zeigen wir dass $A \wedge B \wedge \neg C$ unerfüllbar (widersprüchlich) ist.

Wir betrachten also die Formel:

$$(P \rightarrow (\neg H \rightarrow C)) \wedge (P \rightarrow \neg H) \wedge \neg(P \rightarrow C)$$

Mittels Wahrheitstafel (oder dem später diskutierten Resolutionskalkül) stellen wir fest dass die Formel unerfüllbar ist.

Daher ist der ursprüngliche Schluss auf $P \rightarrow C$ korrekt.

Formalisieren in Aussagenlogik - Beispiel 3

(zum Selbststudium)

Aussage 1: Tom kann am Wochenende nicht beides eine Wanderung und eine Radtour machen.
Aussage 2: Wenn es am Wochenende regnet macht Tom eine Radtour.
Aussage 3: Tom geht am Wochenende Wandern.
Aussage 4 Es regnet nicht am Wochenende.

Aufgabe (Teil 1): Formalisieren Sie die Aussagen mittels aussagenlogischer Formeln. Geben Sie die Bedeutung jeder Aussagenvariablen an.

Atomare Aussagen:
R: „Es regnet am WE“ T: „Tom macht eine Radtour am WE“
W: „Tom macht eine Wanderung am WE“

Aussage 1: $\neg(T \land W)$ oder $(\neg T \lor \neg W)$
Aussage 2: $R \to T$
Aussage 3: W
Aussage 4: $\neg R$

Formalisieren in Aussagenlogik - Beispiel 3

(zum Selbststudium)

Aufgabe (Teil 2): Kann man aus den ersten drei Aussagen auf Aussage 4 schließen? Beweisen Sie Ihre Vermutung mit einer geeigneten Methode aus der Vorlesung (geben Sie auch an welche Methode Sie verwenden).

Wir verwenden die Widerspruchsmethode und betrachten die Konjunktion der drei Aussagen und des negierten Schluss:

$$(\neg T \vee \neg W) \wedge (R \rightarrow T) \wedge W \wedge R$$

Ddiese Formel ist unerfüllbar (Wahrheitstafel). Daher ist der Schluss aus den drei Aussagen korrekt.

Fundamentale Sätze

Satz

Ein Formel F ist genau dann eine Tautologie wenn $\neg F$ unerfüllbar ist.

Beispiel: Tautologie: $(c \vee \neg c)$
unerfüllbar: $\neg(c \vee \neg c) \equiv (\neg c \wedge c)$

Satz

Zwei Formeln F, G sind semantisch äquivalent genau dann wenn F aus G folgt und G aus F folgt.

- $F \equiv G \Leftrightarrow G \models F$ *und* $F \models G$

Satz

Zwei Formeln F, G sind semantisch äquivalent genau dann wenn $F \leftrightarrow G$ eine Tautologie ist

- $F \equiv G \Leftrightarrow \models F \leftrightarrow G$

Es gilt auch:

- $F \models G \Leftrightarrow \models F \rightarrow G$
- $\{F, G\} \models H \Leftrightarrow F \models G \rightarrow H$
- $\{F, G\} \models H \Leftrightarrow \models (F \wedge G) \rightarrow H$

Satz (Kompaktheitssatz)

Eine (unendliche) Menge von aussagenlogischen Formeln ist genau dann erfüllbar, wenn jede endliche Teilmenge erfüllbar ist.

Ersetzungssatz

Beobachtung:

- Jede Teilformel kann durch eine semantisch äquivalente Formel ersetzt werden.

Satz (Ersetzungssatz)

Betrachtet man eine Formel G mit einer Teilformel F_1 und bildet eine neue Formel $G[F_1/F_2]$ indem man (in G) die Teilformel F_1 durch eine semantisch äquivalente Formel F_2 ersetzt, dann gilt $G \equiv G[F_1/F_2]$.

Beispiel: $G = ((\neg a \vee b) \vee c)$

$F_1 = (\neg a \vee b)$ ist semantisch äquivalent zu $F_2 = (a \rightarrow b)$

$G[F_1/F_2] = ((a \rightarrow b) \vee c)$

Ersetzungssatz: $(\neg a \vee b \vee c) \equiv ((a \rightarrow b) \vee c)$

Äquivalenzen der Aussagenlogik

Mit dem Ersetzungssatz kann man äquivalente Formeln verwenden um eine Formel zu vereinfachen. Einige hilfreiche Äquivalenzen:

Idempotenz

- $(F \wedge F) \equiv F$
- $(F \vee F) \equiv F$

Kommutativität

- $(F \wedge G) \equiv (G \wedge F)$
- $(F \vee G) \equiv (G \vee F)$

Assoziativität

- $((F \wedge G) \wedge H) \equiv (F \wedge (G \wedge H))$
- $((F \vee G) \vee H) \equiv (F \vee (G \vee H))$

Absorption

- $(F \wedge (F \vee G)) \equiv F$
- $(F \vee (F \wedge G)) \equiv F$

Distributivität

- $(F \wedge (G \vee H)) \equiv ((F \wedge G) \vee (F \wedge H))$
- $(F \vee (G \wedge H)) \equiv ((F \vee G) \wedge (F \vee H))$

Doppelnegation

- $\neg\neg F \equiv F$

de Morgansche Regeln

- $\neg(F \wedge G) \equiv (\neg F \vee \neg G)$
- $\neg(F \vee G) \equiv (\neg F \wedge \neg G)$

Zusammenfassung & Ausblick

Bis jetzt haben wir Folgendes behandelt:

- Formale Definition von Syntax und Semantik der Aussagenlogik
- Fundamentale Begriffe & Sätze
- Formalisieren in Aussagenlogik

Weiter geht es mit:

- Kalkül der Aussagenlogik
  - Schlussregeln
  - Resolution
  - Hornlogik, Einheitsresolution
- Prädikatenlogik:
  - Syntax
  - Semantik

Subsection 7

Schlussregeln der Aussagenlogik

Der Resolutionskalkül der Aussagenlogik

- Resolution ist ein Verfahren der formalen Logik, um zu Testen ob eine Formel / eine Menge von Formeln widersprüchlich/unerfüllbar ist.
- Diese Herleitung geschieht mittels eines Algorithmus
  $\hookrightarrow$ Maschinengestütztes Beweisen.
- Der Resolutionskalkül arbeitet auf Formeln in KNF (Konjunktiver Normalform).
  $\hookrightarrow$ Der erste Schritt ist die gegebenen Formeln in KNF zu bringen.

Äquivalenzen der Aussagenlogik

Mit dem Ersetzungssatz kann man äquivalente Formeln verwenden um eine Formel zu vereinfachen. Einige hilfreiche Äquivalenzen:

- Idempotenz $(F \wedge F) \equiv F$ und $(F \vee F) \equiv F$
- Kommutativität $(F \wedge G) \equiv (G \wedge F)$ und $(F \vee G) \equiv (G \vee F)$
- Assoziativität $((F \wedge G) \wedge H) \equiv (F \wedge (G \wedge H))$ und
  $((F \vee G) \vee H) \equiv (F \vee (G \vee H))$
- Absorption $(F \wedge (F \vee G)) \equiv F$ und $(F \vee (F \wedge G)) \equiv F$
- Distributivität $(F \wedge (G \vee H)) \equiv ((F \wedge G) \vee (F \wedge H))$
  $(F \vee (G \wedge H)) \equiv ((F \vee G) \wedge (F \vee H))$
- Doppelnegation $\neg\neg F \equiv F$
- de Morgansche Regeln $\neg(F \wedge G) \equiv (\neg F \vee \neg G)$
  $\neg(F \vee G) \equiv (\neg F \wedge \neg G)$

Äquivalenzen der Aussagenlogik

Dank dieser Äquivalenzen können wir Formeln kompakter notieren:

- Klammerung zwischen aufeinander folgender $\wedge$ oder $\vee$ ist nicht notwendig:
  $\hookrightarrow$ Statt $((F \wedge G) \wedge H)$ können wir $(F \wedge G \wedge H)$ schreiben.
- Weiters werden wir Klammern um die ganze Formel nicht notieren:
  $\hookrightarrow$ Statt $(F \wedge G)$ können wir $F \wedge G$ schreiben.

Diese Äquivalenzen sind die Basis für **Normalformen**.

Normalform

Eine **Normalform** ist eine Einschränkung auf der Syntax sodass jede beliebige Formel in eine semantisch äquivalente Formel in Normalform umgewandelt werden kann.

- Vereinfacht die maschinelle Verarbeitung von logischen Formeln. (Viele Algorithmen verarbeiten nur eine bestimmte Normalform)

Konjunktive Normalform

Literal: Unter Literalen verstehen wir

- Atome $x \in \mathit{Var}$ und
- negierte Atome $\neg x$ für $x \in \mathit{Var}$.

Definition

Eine Formel ist in **konjunktiver Normalform (KNF)**, wenn sie eine Konjunktion von Disjunktionen von Literalen ist. F ist also von folgender Form ($L_{i,j}$ Literale):

$$F = (L_{1,1} \vee \cdots \vee L_{1,m_1}) \wedge \cdots \wedge (L_{n,1} \vee \cdots \vee L_{n,m_n})$$

Beispiele:

- $(\neg a \vee b) \wedge (a \vee \neg c \vee d) \wedge \neg b$
- $(\neg a \vee x \vee \neg c \vee d) \wedge (\neg b \vee \neg y)$

Disjunktive Normalform

Definition

Eine Formel ist in **disjunktiver Normalform (DNF)**, wenn sie eine Disjunktion von Konjunktionen von Literalen ist. F ist also von folgender Form ($L_{i,j}$ Literale):

$$F = (L_{1,1} \wedge \cdots \wedge L_{1,m_1}) \vee \cdots \vee (L_{n,1} \wedge \cdots \wedge L_{n,m_n})$$

Beispiele:

- $(\neg a \vee b) \wedge (a \vee \neg c \vee d) \wedge \neg b$ (KNF)
- $(\neg a \wedge b \wedge c) \vee (a \wedge \neg c \wedge d)$ (DNF)
- $\neg a \wedge b \wedge c$ (DNF, KNF)
- $\neg a \vee b \vee c$ (DNF, KNF)
- $(\neg a \vee b \vee c) \wedge (a \vee (b \wedge d))$ ()

Jede Formel kann mit Hilfe der präsentierten Äquivalenzumformungen in eine semantisch äquivalente KNF (DNF) transformiert werden.

Konjunktive und Disjunktive Normalform - Beispiele

Zwei Beispiele wie man eine Formel in KNF umwandeln kann:

$$
\begin{aligned}
&(A \wedge B) \rightarrow C \\
&\neg(A \wedge B) \vee C && \text{(Definition von } \rightarrow\text{)} \\
&(\neg A \vee \neg B) \vee C && \text{(de Morgan)} \\
&\neg A \vee \neg B \vee C && \text{(Assoziativität)}
\end{aligned}
$$

$$
\begin{aligned}
&C \leftrightarrow (A \wedge B) \\
&(\neg C \vee (A \wedge B)) \wedge (C \vee \neg(A \wedge B)) && \text{(Definition von } \leftrightarrow\text{)} \\
&(\neg C \vee (A \wedge B)) \wedge (C \vee (\neg A \vee \neg B)) && \text{(de Morgan)} \\
&(\neg C \vee (A \wedge B)) \wedge (C \vee \neg A \vee \neg B) && \text{(Assoziativität)} \\
&((\neg C \vee A) \wedge (\neg C \vee B)) \wedge (C \vee \neg A \vee \neg B) && \text{(Distributivität)} \\
&(\neg C \vee A) \wedge (\neg C \vee B) \wedge (C \vee \neg A \vee \neg B) && \text{(Assoziativität)}
\end{aligned}
$$

Konjunktive und Disjunktive Normalform - Algorithmus

Diese Schritte lassen sich auch in einen Algorithmus fassen der eine Formel F in eine semantisch äquivalente KNF umzuwandeln.

Algorithmus

1. Ersetze $G \leftrightarrow H$ durch $(G \rightarrow H) \wedge (H \rightarrow G)$
2. Ersetze $G \rightarrow H$ durch $(\neg G \vee H)$
3. Iteriere das Folgende solange wie möglich
   1. Ersetze $\neg\neg G$ durch G
   2. Ersetze $\neg(G \wedge H)$ durch $(\neg G \vee \neg H)$
   3. Ersetze $\neg(G \vee H)$ durch $(\neg G \wedge \neg H)$
4. Iteriere das Folgende solange wie möglich
   1. Ersetze $(G \vee (H \wedge I))$ und $((H \wedge I) \vee G)$ durch $((G \vee H) \wedge (G \vee I))$

„Ersetze“ liest sich als „Ersetze alle Teilformeln der Form“

Der Resolutionskalkül der Aussagenlogik

Der Resolutionskalkül erlaubt die

Unerfüllbarkeit einer aussagenlogischen Formel

zu beweisen.

Viele andere Aufgaben können hierauf zurückgeführt werden:

- Testen ob eine endliche Menge von Formeln $\mathcal{F}$ widersprüchlich ist, dafür testet man die Konjunktion der Formeln auf Unerfüllbarkeit.
- Testen ob eine Formel F aus einer Formelmenge $\mathcal{F}$ folgt.
  Man testet $\mathcal{F} \cup \{\neg F\}$ auf Widersprüchlichkeit.
- Testen ob eine Formel F eine Tautologie ist.
  Man testet $\neg F$ auf Unerfüllbarkeit.

Klauseln von Formeln

Gegeben sei eine Formel F in KNF, also

$$F = (L_{1,1} \vee \cdots \vee L_{1,m_1}) \wedge \cdots \wedge (L_{n,1} \vee \cdots \vee L_{n,m_n})$$

wobei die $L_{i,j}$ Literale sind.

Konjunktion und Disjunktion sind idempotent, kommutativ und assoziativ. Daher können wir eine Formel in KNF auch als Menge von Mengen anschreiben.

Die F zugeordnete **Klauselmenge** $K(F)$ ist

$$K(F) = \{\{L_{1,1}, \ldots, L_{1,m_1}\}, \ldots, \{L_{n,1}, \ldots, L_{n,m_N}\}\}$$

Eine einzelne Menge $\{L_{i,1}, \ldots, L_{i,m_i}\}$ heißt **Klausel** von F.

- Eine Klausel entspricht einer Disjunktion von Literalen

Klauseln von Formeln

- Verschiedene Formeln können dieselbe Klauselmenge besitzen

Beispiel

Die Formeln

$F_1 = (A \vee B) \wedge C$,
$F_2 = (C \vee C) \wedge (A \vee B) \wedge C$ und
$F_3 = C \wedge (B \vee A)$

besitzen alle die Klauselmenge $\{\{A, B\}, \{C\}\}$.

- Eine leere Klausel entspricht dem Wahrheitswert 0 (falsch). (Eine leere Disjunktion ist falsch)
- Eine Klauselmenge die eine leere Klausel enthält hat den Wahrheitswert 0. (Eine Konjunktion ist falsch wenn ein Eintrag falsch ist)
- Eine leere Klauselmenge entspricht dem Wahrheitswert 1. (Eine leere Konjunktion ist wahr)

Resolvente von Klauseln

Idee: Wenn wir Klauseln $\{A, B\}$ und $\{\neg A, C\}$ haben, dann kann nur entweder A oder $\neg A$ wahr sein. Es folgt, dass B oder C wahr sein muss, d.h. die Klausel {B,C}.

Definition

Eine Klausel K_3 heißt **Resolvente** der Klauseln K_1, K_2 wenn es ein Literal L gibt sodass

- $L \in K_1$, $\neg L \in K_2$
- $K_3 = (K_1 \setminus \{L\}) \cup (K_2 \setminus \{\neg L\})$

Wenn K_3 Resolvente von K_1 und K_2 ist, verwenden wir auch folgende graphische Darstellung.

$$\begin{tikzcd} K_1 \arrow[dr, dash] & & K_2 \arrow[dl, dash] \\ & K_3 & \end{tikzcd}$$

Resolvente von Klauseln - Beispiel

Gegeben sei die Klauselmenge $\{\{A, B, \neg C\}, \{A, \neg B, C\}\}$. Zu den Klauseln dieser Klauselmenge gibt es die folgenden Resolventen:

$$\begin{tikzcd}[column sep=small] \{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\ & \{A, C, \neg C\} & \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] \{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\ & \{A, B, \neg B\} & \end{tikzcd}$$

Folgendes ist aber FALSCH:

$$\begin{tikzcd}[column sep=small] \{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\ & \{A\} & \end{tikzcd}$$

Resolutionslemma

Satz (Resolutionslemma)

Für jede Formel F mit Klauselmenge $K(F)$: Ist K_3 eine Resolvente zweier Klauseln $K_1, K_2 \in K(F)$ so ist F semantisch äquivalent zu allen Formeln mit Klauselmenge $K(F) \cup \{K_3\}$.

- Wir können also neue Klauseln hinzufügen ohne die Semantik zu verändern.
- Weiters können wir Klauseln die Tautologien entsprechen (Klauseln die x und $\neg x$ enthalten) weglassen ohne die Semantik zu verändern.

Idee des Resolutionskalküls: Solange Resolventen bilden bis man entweder die leere Klausel enthält oder alle möglichen Resolventen gebildet wurden.

Resolutionskalkül

Sei $K = K(F)$ die Klauselmenge einer Formel:

- $Res(K) = K \cup \{R \mid R \text{ ist Resolvente zweier Klauseln in } K\}$
- $Res^0(K) = K$ und für $n \geq 1$ sei

$$Res^n(K) = Res(Res^{n-1}(K))$$

- $Res^\infty(K) = \bigcup_{n \geq 0} Res^n(K)$

Satz

Eine Formel F ist genau dann unerfüllbar wenn $\emptyset \in Res^\infty(K(F))$.

Hat F nur n Variablen dann hat $Res^\infty(K)$ höchsten 4^n Klauseln.

$\hookrightarrow$ $Res^\infty(K)$ kann mit (maximal) 4^n Iterationen berechnet werden.

Resolutionskalkül - Erfüllbarkeitstest

Gegeben sei eine Formel in KNF:

1. Bilde Klauselmenge $K(F)$
2. Berechne $Res^1(K), Res^2(K), \ldots, Res^n(K)$ bis
$$\emptyset \in Res^n(K) \text{ oder } Res^{n-1}(K) = Res^n(K).$$
3. Wenn $\emptyset \in Res^n(K)$ „F unerfüllbar“

   anderenfalls „F erfüllbar“.

Formalisieren in Aussagenlogik - Beispiel 2 VO1

Aussage 1: Wenn der Geiger das Konzert gibt, werden viele kommen, wenn die Preise nicht zu hoch sind.
Aussage 2: Wenn der Geiger das Konzert gibt, werden die Preise nicht zu hoch sein.
Schluss: Daher werden, falls der Geiger das Konzert bestreitet, viele kommen.

Wir wollen wissen ob der Schluss korrekt ist.

Atomare Aussagen:
P: „der Geiger gibt das Konzert"
C: „viele werden kommen"
H: „die Preise sind zu hoch"

Aussage 1: $P \to (\neg H \to C)$
Aussage 2: $P \to \neg H$
Schluss: $P \to C$

Wir wollen wissen ob $P \to C$ aus $P \to (\neg H \to C)$ und $P \to \neg H$ folgt.

Resolutionskalkül - Beispiel

Wir betrachten die Formel aus dem Geiger-Beispiel:

$$(P \to (\neg H \to C)) \land (P \to \neg H) \land \neg(P \to C)$$

1.Schritt: Wir bringen die Formel in KNF.

$$\begin{array}{rl} (\neg P \lor (H \lor C)) \land (\neg P \lor \neg H) \land \neg(\neg P \lor C) & \text{Definition von } \to \\ (\neg P \lor H \lor C) \land (\neg P \lor \neg H) \land P \land \neg C & \text{de Morgan, Assoziativität} \end{array}$$

2.Schritt: Bilden der Klauselmengen

$$K(F) = \{\{\neg P, H, C\}, \{\neg P, \neg H\}, \{P\}, \{\neg C\}\}$$

Resolutionskalkül - Beispiel

3.Schritt: Resolventen berechnen

$$Res^0(K) = \{\{\neg P, H, C\}, \{\neg P, \neg H\}, \{P\}, \{\neg C\}\}$$
$$Res^1(K) = Res^0(K) \cup \{\{\neg P, C\}, \{H, C\}, \{\neg P, H\}, \{\neg H\}\}$$
$$Res^2(K) = Res^1(K) \cup \{\{C\}, \{\neg P\}, \{H\}\}$$
$$Res^3(K) = Res^2(K) \cup \{\{\}\}$$

Da wir die leere Klausel erreicht haben ist die Formel unerfüllbar.

Resolutionskalkül - Beispiel

Oft müssen wir nicht alle Resolventen berechnen um einen Widerspruch (die leere Klausel) zu finden:

$$
\begin{tikzcd}
\{\neg P, H, C\} \arrow[dr, dash] & & \{\neg P, \neg H\} \arrow[dl, dash] & & \{P\} \arrow[ddl, dash] & & \{\neg C\} \arrow[dddl, dash] \\
& \{\neg P, C\} \arrow[drr, dash] & & & & & \\
& & & \{C\} \arrow[drr, dash] & & & \\
& & & & & \emptyset &
\end{tikzcd}
$$

Aber: Um zu zeigen dass eine Formel widerspruchsfrei ist muss man alle Resolventen berechnen!!!

Subsection 9

Einheitsresolution

Einheitsresolution

Eine **Einheitsklausel** ist eine Klausel die nur aus einem (positiven oder negativen) Literal besteht.

Beobachtung: Enthält die Klauselmenge

- eine positive Einheitsklausel $\{x\}$ so muss x jedenfalls auf 1 gesetzt werden
- eine negative Einheitsklausel $\{\neg x\}$ so muss x jedenfalls auf 0 gesetzt werden

um die Formel zu erfüllen.

Einheitsresolution (unit propagation)

Formel F in KNF, $K = K(F)$

- Solange die Klauselmenge K eine Einheitsklausel enthält wähle eine Einheitsklausel $\{L\}$
  - Entferne alle Klauseln in K die L enthalten
  - Lösche $\neg L$ aus allen Klauseln in K
- Wenn $\emptyset \in K$ dann ist die Formel F unerfüllbar.

Einheitsresolution

Sei K eine Klauselmenge dann schreiben wir $ERes(K)$ für die vereinfachte Klauselmenge die der Einheitsresolution-Algorithmus erzeugt.

Satz (Erfüllbarkeitsäquivalenz)

Eine Klauselmenge K ist genau dann erfüllbar wenn ihre Vereinfachung $ERes(K)$ durch Einheitsresolution erfüllbar ist.

- Wenn $\emptyset \in ERes(K)$ dann ist die Formel F unerfüllbar.

Einheitsresolution und allgemeine Klauselmengen

Gegeben: Klauselmenge $K(F)$.

Ziel: Testen ob die Klauselmenge $K(F)$ erfüllbar ist.

Idee: Kombiniere Resolution mit Einheitsresolution

Verfahren: Wandle den Erfüllbarkeitstest mittels Resolution wie folgt ab

- Zu Beginn starte Einheitsresolution um die Klauselmenge zu verkleinern.
- Wurde die leere Klausel berechnet, wird ist $K(F)$ unerfüllbar; sonst starte Resolution auf der verkleinerten Klauselmenge.
- Wann immer Resolution eine Einheitsklausel hinzufügt, starte Einheitsresolution um die Klauselmenge zu verkleinern bevor mit Resolution fortgesetzt wird.

Einheitsresolution und allgemeine Klauselmengen - Beispiel

Beispiel

Formel $F = a \wedge (\neg a \vee b \vee c) \wedge (b \leftrightarrow c) \wedge (\neg a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c)$
Klauselmenge
$K(F) = \{\{a\}, \{\neg a, b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg a, \neg b, \neg c\}, \{a, b, c\}\}$
mit Einheitsklausel $\{a\}$

Einheits-Resolution mit $\{a\}$ ergibt: $\{\{b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg b, \neg c\}\}$

Ein Resolution-Schritt ergibt: $\{\{b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg b, \neg c\}, \{b\}\}$

Einheits-Resolution mit $\{b\}$ ergibt: $\{\{c\}, \{\neg c\}\}$

Einheits-Resolution mit $\{\neg c\}$ ergibt: $\{\{\}\}$

Die Formel F ist also unerfüllbar.

Hornlogik

Die Hornlogik ist eine Einschränkung der Aussagenlogik, d.h. es sind nur Formeln einer speziellen Form erlaubt.

- Benannt nach dem Logiker Alfred Horn.
- Kann mit speziellen Resolutions-Varianten sehr effizient ausgewertet werden.
  $\hookrightarrow$ Einheitsresolution, SLD-Resolution
- Dient als Grundlage der logischen Programmierung (Prolog).
- Bedeutung in der Komplexitätstheorie.

Hornlogik

Definition

Eine Klausel heißt **Hornklausel** wenn sie höchstens ein positives Literal enthält. Eine Formel F in KNF heißt **Hornformel** wenn $K(F)$ nur aus Hornklauseln besteht.

Beispiel für eine Hornformel:

- $(a \vee \neg b \vee \neg c) \wedge (\neg b \vee \neg c) \wedge c$

Unterschiedliche Arten von Hornklauseln:

- Tatsachenklauseln/Fakten: nur ein positives Literal, z.B. c
- Regeln: ein positives Literal, z.B. $(a \vee \neg b \vee \neg c)$ $\big(\equiv (b \wedge c) \rightarrow a\big)$
- Zielklausel: nur negative Literale, z.B. $(\neg b \vee \neg c)$

Einheitsresolution

Sei K eine Klauselmenge dann schreiben wir $ERes(K)$ für die vereinfachte Klauselmenge die der Einheitsresolution-Algorithmus erzeugt.

Satz (Erfüllbarkeitsäquivalenz)

Eine Klauselmenge K ist genau dann erfüllbar wenn ihre Vereinfachung $ERes(K)$ durch Einheitsresolution erfüllbar ist.

- Wenn $\emptyset \in ERes(K)$ dann ist die Formel F unerfüllbar.
- Wenn K eine **Horn-Formeln** ist gilt auch:
  - Wenn $\emptyset \notin ERes(K)$ dann ist die Formel erfüllbar.

Satz

Eine Horn-Formel ist genau dann erfüllbar wenn in $\emptyset \notin ERes(K)$.

Hornlogik/Einheitsresolution - Beispiel 1

Beispiel

Formel $F = a \wedge b \wedge ((a \wedge b) \rightarrow c) \wedge (d \rightarrow e)$
Klauselmenge $K(F) = \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg d, e\}\}$
mit zwei Einheitsklausel $\{a\}, \{b\}$

$$\begin{array}{rl}
\{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg d, e\}\} & \text{Resolviere } \{a\} \\
\{\{b\}, \{\neg b, c\}, \{\neg d, e\}\} & \text{Resolviere } \{b\} \\
\{\{c\}, \{\neg d, e\}\} & \text{Resolviere } \{c\} \\
\{\{\neg d, e\}\} &
\end{array}$$

Die Formel F ist erfüllbar.

Hornlogik/Einheitsresolution - Beispiel 2

Beispiel

Formel $F = a \wedge b \wedge ((a \wedge b) \rightarrow c) \wedge (\neg a \vee \neg b) \wedge (d \rightarrow e)$
Klauselmenge $K(F) = \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg a, \neg b\}, \{\neg d, e\}\}$
Zwei Einheitsklausel $\{a\}, \{b\}$

$$\begin{aligned} \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg a, \neg b\}, \{\neg d, e\}\} &\quad \text{Resolviere } \{a\} \\ \{\{b\}, \{\neg b, c\}, \{\neg b\}, \{\neg d, e\}\} &\quad \text{Resolviere } \{b\} \\ \{\{c\}, \{\}, \{\neg d, e\}\} & \end{aligned}$$

Da wir die leere Klausel erhalten haben ist die Formel F unerfüllbar.

Hornlogik/Einheitsresolution - Beispiel 3

(zum Selbststudium)

Betrachte die Formel

$$F = (\neg T \vee \neg W) \wedge (R \rightarrow T) \wedge W \wedge R$$

Durch Auflösen der Inklusion $(R \rightarrow T)$ zu $(\neg R \vee T)$ erhalten wir

$$K(F) = \{\{\neg T, \neg W\}, \{\neg R, T\}, \{W\}, \{R\}\}$$

$$\begin{array}{rl}
\{\{\neg T, \neg W\}, \{\neg R, T\}, \{W\}, \{R\}\} & \text{Resolviere } \{W\} \\
\{\{\neg T\}, \{\neg R, T\}, \{R\} & \text{Resolviere } \{\neg T\} \\
\{\{\neg R\}, \{R\}\} & \text{Resolviere } \{\neg R\} \\
\{\{\}\} &
\end{array}$$

Da wir die leere Klausel erhalten haben ist die Formel F unerfüllbar.

Subsection 11

Limitierungen von Aussagenlogik

Limitierungen von Aussagenlogik

Es gibt logische Zusammenhänge die sich mit Aussagenlogik nicht abbilden lassen.

Beispiel

Aussage 1: Fritz fährt Ski
Aussage 2: Uli fährt Ski

Obwohl beide Aussagen augenscheinlich von gleicher Natur sind, kann Aussagenlogik die gemeinsame Struktur nicht modellieren.

Beispiel

Aussage 1: Jeder Mensch ist wertvoll.
Aussage 2: Hans ist ein Mensch.

Wir würden also gerne schließen „Hans ist wertvoll“. Das ist aber in Aussagenlogik nicht möglich.

Limitierungen von Aussagenlogik

Limitierungen von Aussagenlogik

- Aussagen werden als Atome genutzt und nicht weiter analysiert
  $\hookrightarrow$ oft stecken mehrere Informationen in einer atomaren Aussage
- Die innere Struktur einer Aussage geht verloren.
- Es ist unmöglich/schwer auszudrücken, dass
  - gewisse Beziehungen zwischen Objekten gelten;
  - etwas für alle Objekte gilt; oder
  - es ein Objekt mit einer Eigenschaft geben muss.

Wir benötigen also eine **ausdrucksstärkere Logik**.

Prädikatenlogik

Zusammenfassung & Ausblick

Bisher

- Worum geht es in der Logik
- Aussagenlogik:
  - Formale Definition von Syntax und Semantik der Aussagenlogik
  - Fundamentale Sätze
  - Formalisieren und Schließen in der Aussagenlogik
  - Schlussregeln
  - Resolutionskalkül der Aussagenlogik
  - Hornlogik und Einheitsresolution

Nächste Einheit:

- Ausblick Prädikatenlogik
- Einführung Prädikatenlogik
- Formale Syntax der Prädikatenlogik
- Formale Semantik der Prädikatenlogik
- Formalisieren in Prädikatenlogik