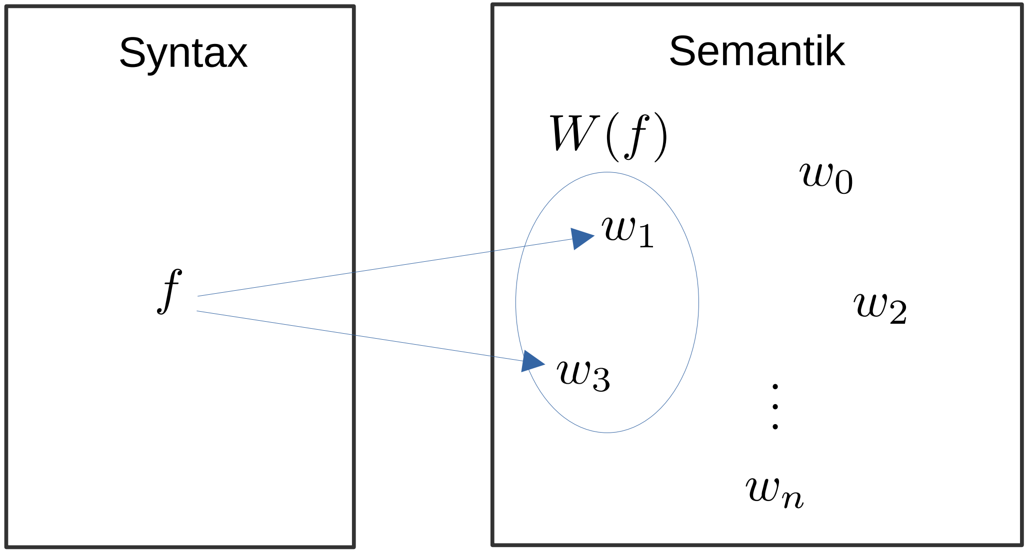
Syntax auf Semantik

Wir können eine Teilmenge von Welten finden, die eine bestimmte Formel erfüllen.
Sei $W(f)$ eine Menge von Formeln, für die $I(f,w)=1$ ist.

Beispiel: $W(f)=\{w_1,w_3\}$



Wissensbasis

Eine **Wissensbasis** WB ist eine Menge von Formeln, die ihre Konjunktion/Schnittmenge darstellen. Die Formeln in WB sind auch als **Fakten** bekannt.

Eine Wissensbasis, die aus einer Reihe von Formeln besteht, kann als Beschränkung für ein Problem angesehen werden. Ähnlich wie wir eine Menge von Welten für eine einzelne Formel erhalten können, können wir eine Menge von Welten $W(\mathrm{WB})$ definieren, die die von der Wissensbasis auferlegten Beschränkungen erfüllen.

$$W(\mathrm{WB}) = \bigcap_{f \in \mathrm{WB}} W(f)$$

Beispiel - Spracherkennung

Ein Chatbot sollte in der Lage sein, die Sprache des Nutzerin oder des Nutzers anhand der Position des Substantivs in einer Frage automatisch zu erkennen. Steht das Substantiv $\mathbf{N}$ an Position 1 (wahr), schaltet der Chatbot in den internationalen Modus $\mathbf{I}$. Steht es an Position 0 (falsch), bleibt er im lokalen Modus (Deutsch).

Position:	0	1	
Can I:	[Verb]	$\mathbf{N}$	?
Kann ich:	$\mathbf{N}$	[Verb]	?

Bekannte Fakten

$$\mathrm{WB} = \{\mathbf{N} \rightarrow \mathbf{I}\}$$

Beispiel - Spracherkennung

Wenn der Benutzer eine Frage auf Englisch eingibt, steht das Nomen an Position 1, wodurch $\mathbf{N}$ wahr wird. Wir erhalten also die neue bekannte Fakten $\mathbf{N}$ und fügen sie der Wissensbasis hinzu.

Position:		0	1	
Can I		cook	**spaghetti**	?

Bekannte Fakten

$$\text{WB} = \{\mathbf{N}, \mathbf{N} \rightarrow \mathbf{I}\}$$

Wir nehmen $\mathbf{I}$ nicht in die Wissensbasis auf, da wir nur Basisfakten und Implikationen in die WB aufnehmen. Die Wahrheit von $\mathbf{I}$ wird nicht unabhängig behauptet, sondern muss mithilfe von $\mathbf{N} \rightarrow \mathbf{I}$ abgeleitet werden. Wenn die WB $\mathbf{I}$ explizit enthielte, würde dies bedeuten, dass $\mathbf{I}$ immer wahr ist, was nicht unbedingt der Fall ist.

Beispiel - Spracherkennung

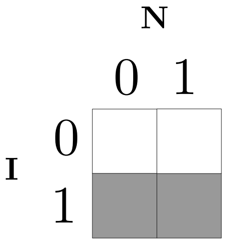
$W(\mathrm{WB})$ zu finden bedeutet, zu erkennen, dass es sich um eine Konjunktion aller Welten handelt, die die durch die Fakten in der Wissensbasis definierten Einschränkungen erfüllen:

$$W(\mathrm{WB}) = \bigcap_{f \in \mathrm{WB}} W(f) = W(\mathbf{N}) \bigcap W(\mathbf{N} \rightarrow \mathbf{I}))$$

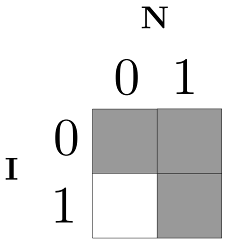
Beispiel - Spracherkennung

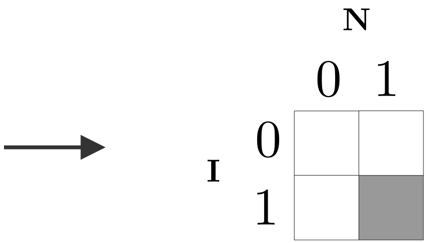
Welten	$\mathbf{N}$	$\mathbf{N} \to \mathbf{I}$
$w_0 = \{\mathbf{N} : 0, \mathbf{I} : 0\}$	0	1
$w_1 = \{\mathbf{N} : 0, \mathbf{I} : 1\}$	0	1
$w_2 = \{\mathbf{N} : 1, \mathbf{I} : 0\}$	1	0
$w_3 = \{\mathbf{N} : 1, \mathbf{I} : 1\}$	1	1

$W(\mathbf{N})$



$W(\mathbf{N} \to \mathbf{I})$





Beispiel - Spracherkennung

Die Menge der Welten, die die Bedingungen in WB erfüllen, besteht nur aus w_3:

$$W(\mathrm{WB}) = \bigcap_{f \in \mathrm{WB}} W(f) = W(\mathbf{N}) \bigcap W(\mathbf{N} \to \mathbf{I})) = \{w_3\}$$

Wie können wir beweisen, dass dies mathematisch wahr ist? Ein Ansatz könnte darin bestehen, logische Äquivalenzen zu verwenden, um die Konjunktion der Formeln in WB zu **vereinfachen**.

Äquivalenzgesetze

1. Identität	$p \wedge 1 \equiv p$	$p \vee 0 \equiv p$
2. Dominanz	$p \wedge 0 \equiv 0$	$p \vee 1 \equiv 1$
3. Idempotenz	$p \wedge p \equiv p$	$p \vee p \equiv p$
4. Doppelnegation	$\neg(\neg p) \equiv p$	
5. Kommutativ	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
6. Assoziativ	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
7. Distributiv	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
8. De-Morgan-Gesetze	$\neg(p \wedge q) \equiv \neg q \vee \neg p$	$\neg(p \vee q) \equiv \neg q \wedge \neg p$
9. Absorptions	$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
10. Implikation	$p \rightarrow q \equiv \neg p \vee q$	
11. Kontraposition	$p \rightarrow q \equiv \neg q \rightarrow p$	

Ansatz

Gegeben: $W(\mathbf{N}) \bigcap W(\mathbf{N} \rightarrow \mathbf{I}))$

$\Longrightarrow \mathbf{N} \wedge (\mathbf{N} \rightarrow \mathbf{I})$

$\equiv \mathbf{N} \wedge (\neg\mathbf{N} \vee \mathbf{I})$ Gesetz der Implikation

$\equiv (\mathbf{N} \wedge \neg\mathbf{N}) \vee (\mathbf{N} \wedge \mathbf{I})$ Distributiv (UND)

$\equiv 0 \vee (\mathbf{N} \wedge \mathbf{I})$

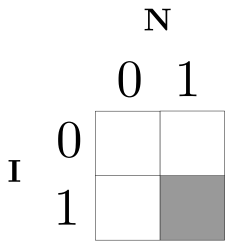
$\equiv (0 \vee \mathbf{N}) \wedge (0 \vee \mathbf{I})$ Distributiv (ODER)

$\equiv (\mathbf{N} \wedge \mathbf{I})$ Identität

Ansatz

Die Konjunktion der Formeln in WB entspricht w_3 in unserer Wahrheitstabelle.

$\mathbf{N}$	$\mathbf{I}$	$(\mathbf{N} \wedge \mathbf{I})$
0	0	0
0	1	0
1	0	0
1	1	1



Inferenzregeln (Schlussregeln)

Können wir aus bekannten Fakten (Formeln) neue logische Formeln ableiten? Ja. Für n Formeln f kann eine neue Formel g abgeleitet werden.

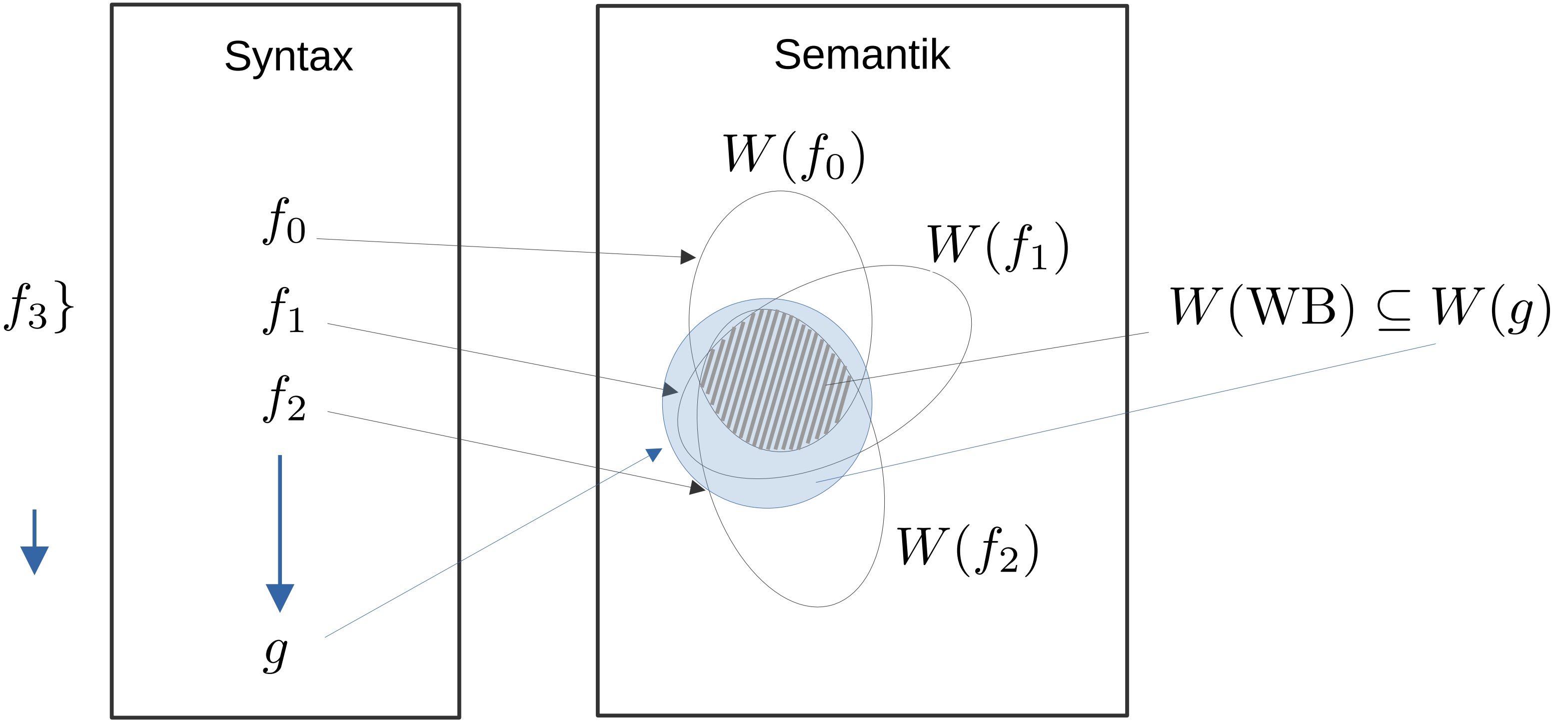
$$\frac{f_0, f_1 \cdots, f_n}{g}$$

Während W(WB) eine Menge von Welten in der Semantik ist, können wir eine Formel g ableiten, die eine Menge von Welten W(g) hat, die eine Obermenge von W(WB) ist.

Inferenzregeln - Konzept

$\mathrm{WB} = \{f_0, f_1, f_3\}$

Inferenzregeln



Inferenzregel - Modus Ponens

Wenn eine Aussage der Form "Wenn p, dann q" wahr ist und p wahr ist, dann ist auch q wahr.

$$\frac{p,\ \ p \to q}{q}$$

Beispiel: Wenn wir wissen es regnet Regen, und auch “wenn es regnet, dann ist der Boden nass” $\mathrm{Regen} \to \mathrm{Nass}$, daraus lässt sich schließen, dass ser Boden nass ist.

$$\frac{\mathrm{Regen},\ \ \mathrm{Regen} \to \mathrm{Nass}}{\mathrm{Nass}}$$

Inferenzregel - Resolution

Die Resolution ist eine Inferenz(Schluss)regel, die neue Klauseln ableitet, indem sie komplementäre **Literale** aus zwei **Klauseln** in konjunktiver Normalform (**KNF**) eliminiert.

$$\frac{(p \lor q) \land (\neg q \lor r)}{(p \lor r)}$$

- **Literal**: eine atomare Formel oder sein Komplement $p \text{ bzw. } \neg p$
- **Klausel**: eine Disjunktion von Literalen $p \lor q$
- **KNF**: eine Konjunktion von Klauseln $(p \lor q) \land (\neg q \lor r)$

Inferenzalgorithmus

Mit Inferenzalgorithmen werden die Inferenzen in einem wissensbasierten System berechnet. Für unsere Wissensbasis können wir einen Algorithmus implementieren, der die beiden bis hierher behandelten Inferenzregeln iterativ anwendet:

1. Alle neuen bekannten Fakten in die Wissensbasis aufnehmen
2. Bestimmen Sie, ob wir neue Formeln aus allen bekannten Fakten ableiten können
3. Wenn **ja**, *führen Sie Inferenzregeln aus*, um die neue Formel abzuleiten, fügen Sie sie als neue bekannte Tatsache wieder in die Wissensbasis ein und wiederholen Sie den Vorgang. Wenn **nein**, *beenden Sie die Schleife*.

Beispiel - Spracherkennung

Ein Chatbot sollte in der Lage sein, die Sprache der Nutzerin oder des Nutzers anhand der Position des Substantivs in einer Frage automatisch zu erkennen. Steht das Substantiv N an Position 1 (wahr), schaltet der Chatbot in den internationalen Modus I. Steht es an Position 0 (falsch), bleibt er im lokalen Modus (Deutsch). Wenn I wahr ist, ist dann ist die Sprache S 1 (wahr) und als Englisch gesetzt. Wenn die Sprache S 0 (falsch) ist, ist die Sprache als Default (Deutsch) erkannt.

Position:	0	1	
Can I:	[Verb]	$\mathbf{N}$	?
Kann ich:	$\mathbf{N}$	[Verb]	?

<u>Bekannte Fakten</u>

$$\mathrm{WB} = \{\mathbf{N} \to \mathbf{I}\}$$

$$\mathrm{WB} = \{\mathbf{I} \to \mathbf{S}\}$$

Ansatz

1. Bekannte Fakten in WB ermitteln

$$\mathrm{WB} = \{\mathbf{N} \to \mathbf{I}, \mathbf{I} \to \mathbf{S}\}$$

2. Wir erhalten als Eingabe "Can I cook spaghetti?"

3. Da $\mathbf{N}$ wahr ist, fügen wir es der WB hinzu

$$\mathrm{WB} = \{\mathbf{N}, \mathbf{N} \to \mathbf{I}, \mathbf{I} \to \mathbf{S}\}$$

4.Wir verwenden *Modus Ponens*, um $\mathbf{I}$ zu erhalten und fügen es wieder in die WB ein

$$\frac{\mathbf{N},\ \ \mathbf{N} \to \mathbf{I}}{\mathbf{I}} \quad \longrightarrow \quad \mathrm{WB} = \{\mathbf{N}, \mathbf{N} \to \mathbf{I}, \mathbf{I}, \mathbf{I} \to \mathbf{S}\}$$

5. In der zweiten Iteration verwenden wir *Modus Ponens*, um $\mathbf{S}$ zu erhalten, und setzen die Sprache auf Englisch.

$$\frac{\mathbf{I},\ \ \mathbf{I} \to \mathbf{S}}{\mathbf{S}} \quad \longrightarrow \quad \mathrm{WB} = \{\mathbf{N}, \mathbf{N} \to \mathbf{I}, \mathbf{I}, \mathbf{I} \to \mathbf{S}, \mathbf{S}\}$$

Alternativansatz - Resolution

Wir können die Inferenzregel Resolution verwenden, um die Formeln in der WB zu vereinfachen und die Anzahl der Iterationen in unserem Algorithmus zu reduzieren.

1. WB ist eine *Menge von Formeln* (Syntax), die ihre Konjunktion/Schnittmenge darstellen.

$$(\mathbf{N} \to \mathbf{I}) \land (\mathbf{I} \to \mathbf{S})$$

2. Gesetz der Implikation

$$(\neg \mathbf{N} \lor \mathbf{I}) \land (\neg \mathbf{I} \lor \mathbf{S})$$

3. Resolutionsverfahren

$$\frac{(\neg \mathbf{N} \lor \mathbf{I}) \land (\neg \mathbf{I} \lor \mathbf{S})}{(\neg \mathbf{N} \lor \mathbf{S})}$$

4. Gesetz der Implikation

$$(\mathbf{N} \to \mathbf{S})$$

5. Neue WB

$$\mathrm{WB} = \{\mathbf{N}, \mathbf{N} \to \mathbf{S}\}$$

6.Wir verwenden *Modus Ponens*, um $\mathbb{S}$ zu erhalten und setzen die Sprache auf Englisch

$$\frac{\mathbf{N},\ \ \mathbf{N} \to \mathbf{S}}{\mathbf{S}} \quad \longrightarrow \quad \mathrm{WB} = \{\mathbf{N}, \mathbf{N} \to \mathbf{S}, \mathbf{S}\}$$