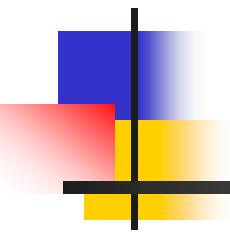


Hauptseminar: Linguistic Inference and Textual Entailment
Leitung: Prof. Dr. Manfred Pinkal
Sommersemester 2007



Linguistic Inference
with
Model Generation



Aufbau

- 1. Einleitung
- 2. Systemaufbau
- 3. Linguistische Anwendung
- 4. Zusammenfassung



1. Einleitung

Thema:

- Automatisierte Bedeutungserfassung

Inhalt:

- Funktionsweise und Eigenschaften eines Systems, welches sich für diese Aufgabe gut zu eignen scheint
- Linguistische Phänomene, die sich mit Hilfe von solch einem System bearbeiten lassen



1.1 Motivation

Ein Beispieltext:

- *(1) Wenn ich Pommes esse, dann nehme ich mir weder Salat, noch Fisch.*
- *(2) Ohne Pommes esse ich Fisch und Salat nicht zusammen.*
- *(3) Ein Dessert gibt es immer genau dann, wenn ich keine Pommes habe.*
- *(4) Mein Essen besteht aus mindestens zwei Komponenten.*
- *(5) Salat und Dessert alleine reichen mir nicht aus.*

Semantische Repräsentation:

$$(1) P \rightarrow \neg (S \vee F)$$

$$(2) \neg P \rightarrow \neg (S \wedge F)$$

$$(3) D \leftrightarrow \neg P$$

$$(4) (P \wedge S) \vee (P \wedge F) \vee (P \wedge D) \vee (S \wedge F) \vee (S \wedge D) \vee (F \wedge D)$$

$$(5) S \wedge D \rightarrow P \vee F$$



1.1 Motivation

Es gibt generell 2 Möglichkeiten:

- der Text enthält einen Widerspruch
- man kann eine mögliche Mahlzeit angeben

In diesem Fall ist die Formelmenge für $\{F, D, \neg S, \neg P\}$ erfüllbar:

- „*ich esse immer nur Fisch und Dessert*“



2. Systemaufbau

Ein System, welches uns bei der Interpretation natürlicher Sprache behilflich sein soll, muss mindestens folgende Bestandteile haben:

- 1. Werkzeuge für die Semantikkonstruktion (Spracherkenner, Parser usw.)
- 2a. Theorembeweiser
- 2b. Modellerzeuger



2.1 Komponenten

- 1. der Diskursabschnitt muss in eine für automatische Verarbeitung geeignete Form überführt werden, z.B. ein Logikkalkül.
- 2a. Theorembeweiser beweist die Nicherfüllbarkeit (Inkonsistenz) einer Formelmengemenge Φ
- 2b. Modellerzeuger beweist dagegen die Erfüllbarkeit (Konsistenz) einer Formelmengemenge Φ und erzeugt eine Modellstruktur die Φ wahr macht



2.2 Modelleigenschaften

Ein Modell $M = \langle D, I \rangle$ besteht aus einem nicht-leeren Universum D und einer Interpretationsfunktion I .

- 1. Es gilt die Annahme zur Weltabgeschlossenheit. D.h. alle in der Modellstruktur nicht explizit angegebenen Fakten werden als falsch behandelt.



2.2 Modelleigenschaften

- 2. Es soll keine andere Modellstruktur $M' = \langle D', I' \rangle$ geben, so dass $I' \subseteq I$. In anderen Worten es soll eine minimale Struktur sein, die keine unnötigen Individuen und Relationen enthält.

Vorsicht:

- Man darf das System mit Wissen nicht unterversorgen



2.2 Modelleigenschaften

- (1) *Bugs likes a white female rabbit.*
 $\exists x[rabbit(x) \wedge white(x) \wedge female(x) \wedge like(bugs, x)]$
- Ein Modell für (1): $\{rabbit(bugs), white(bugs), female(bugs), like(bugs, bugs)\}$
- In diesem Fall ist das generierte minimale Modell kleiner als die bevorzugte Interpretation des Satzes.
- Es fehlt das Weltwissen , dass *Bugs* ein männlicher Name ist: $rabbit(bugs) \wedge \neg female(bugs)$



2.2 Modelleigenschaften

- (2) *Mia doesn't know Vincent, every man knows a woman, Butch is a man, Vincent is a man, Mia is a woman, and men are disjoint from women.*

$\neg \text{know}(\text{mia}, \text{vincent}) \wedge \forall x(\text{man}(x) \rightarrow \exists y(\text{woman}(y) \wedge \text{know}(x, y)))$

$\wedge \text{man}(\text{butch}) \wedge \text{man}(\text{vincent}) \wedge \text{woman}(\text{mia}) \wedge \forall x(\text{man}(x) \rightarrow \neg \text{woman}(x))$

- $D = \{d1, d2\}$
- $I(\text{mia}) = d1, I(\text{butch}) = d2, I(\text{vincent}) = d2,$
 $I(\text{man}) = \{d2\}, I(\text{woman}) = \{d1\}, I(\text{know}) = \{(d1, d2)\}$

3. Linguistische Anwendung



Modellerzeuger sind ein vielversprechendes Werkzeug für linguistische Anwendungen, weil:

- Sie mit sehr ausdrucksstarken formalen Sprachen (wie λ -Typ-Sprache) arbeiten
- Ihre Modelle sehr effiziente flache Strukturen haben



3.1 Anaphoric Linking

- (3) A rabbit sleeps. The rabbit dreams.
- Semantik von 'a': $\lambda F \lambda G \exists x (F(x) \wedge G(x))$
- Semantik von 'the': $\lambda F \lambda G \exists x (\forall y (F(y) \leftrightarrow x = y) \wedge G(x))$
- Semantik von dem gesamten Beispiel:

$\lambda F \lambda G \exists x (F(x) \wedge G(x))(rabbit)(sleep) \wedge$

$\lambda F \lambda G \exists x (\forall y (F(y) \leftrightarrow x = y) \wedge G(x))(rabbit)(dream)$

$\xrightarrow{\beta} \exists x_1 (rabbit(x_1) \wedge sleep(x_1)) \wedge \exists x_2 (rabbit(x_2) \wedge dream(x_2))$

- $\{rabbit(c_1), sleep(c_1), dream(c_1)\},$



3.2 Bridging

- (4) *A rabbit dreams. The tail is twitching.*
- + Weltwissen: *Every rabbit has a unique tail. Tails and rabbits are different entities.*
- Semantik von 'every': $\lambda F \lambda G \forall x (F(x) \rightarrow G(x))$
- Semantik von 'not any': $\lambda F \lambda G \neg \exists x (F(x) \wedge G(x))$

3.2 Bridging

- Semantik von dem gesamten Beispiel:

$$\begin{aligned} & \lambda F \lambda G \exists x (F(x) \wedge G(x))(rabbit)(dream) \wedge \\ & \lambda F \lambda G \exists x (\forall y (F(x) \leftrightarrow x = y) \wedge G(x))(tail)(twich) \wedge \\ & \lambda F \lambda G \forall x (F(x) \rightarrow G(x))(rabbit)(\lambda x \exists y (tail(y) \wedge of(x)(y))) \wedge \\ & \lambda F \lambda G \neg \exists x (F(x) \wedge G(x))(rabbit)(tail) \end{aligned}$$
$$\begin{aligned} \xrightarrow{\beta} & \exists x (rabbit(x) \wedge dream(x)) \wedge \\ & \exists x (\forall y (tail(x) \leftrightarrow x = y) \wedge twich(x)) \wedge \\ & \forall x (rabbit(x) \rightarrow \lambda x \exists y (tail(y) \wedge of(x)(y))(x)) \wedge \\ & \neg \exists x (rabbit(x) \wedge tail(x)) \end{aligned}$$
$$\begin{aligned} \xrightarrow{\beta} & \exists x (rabbit(x) \wedge dream(x)) \wedge \\ & \exists x (\forall y (tail(x) \leftrightarrow x = y) \wedge twich(x)) \wedge \\ & \forall x (rabbit(x) \rightarrow \exists y (tail(y) \wedge of(x)(y)) \wedge \\ & \neg \exists x (rabbit(x) \wedge tail(x)) \end{aligned}$$



3.2 Bridging

- $\{\text{rabbit}(c_1), \text{tail}(c_2), \text{twitch}(c_2), \text{sleep}(c_1), \text{of}(c_2, c_1)\}$
- (4) "A rabbit dreams. The tail is twitching."
- (4') "A rabbit dreams. The tail of that rabbit is twitching"

3.3 Accommodation

- *(5) The rabbit eats the carrot.*

$$\lambda F \lambda G \exists x_1 (\forall y_1 (F(x_1) \leftrightarrow x_1 = y_1) \wedge$$
$$G(x_1))(rabbit)(\lambda F \lambda G \exists x_2 (\forall y_2 (F(x_2) \leftrightarrow x_2 = y_2) \wedge G(x_2)))(carrot)(\lambda z_1 \lambda z_2 (eat(z_1, z_2)))$$
$$\xrightarrow{\beta} \exists x_1 (\forall y_1 (rabbit(x_1) \leftrightarrow x_1 = y_1) \wedge$$
$$\lambda F \lambda G \exists x_2 (\forall y_2 (F(x_2) \leftrightarrow x_2 = y_2) \wedge G(x_2)))(carrot)(\lambda z_1 \lambda z_2 (eat(z_1, z_2))(x_1))$$
$$\xrightarrow{\beta} \exists x_1 (\forall y_1 (rabbit(x_1) \leftrightarrow x_1 = y_1) \wedge$$
$$\exists x_2 (\forall y_2 (carrot(x_2) \leftrightarrow x_2 = y_2) \wedge \lambda z_1 \lambda z_2 (eat(z_1, z_2))(x_1)(x_2)))$$
$$\xrightarrow{\beta} \exists x_1 (\forall y_1 (rabbit(x_1) \leftrightarrow x_1 = y_1) \wedge \exists x_2 (\forall y_2 (carrot(x_2) \leftrightarrow x_2 = y_2) \wedge (eat(x_1, x_2))))$$

- $\{rabbit(c_1), carrot(c_2), eat(c_1, c_2)\}$



3.4 Abductive reasoning

- *„Abduktion ist der Vorgang, in dem eine erklärende Hypothese gebildet wird“*
(Charles Sanders Peirce, Pragmatism and Pragmaticism (1934)).
- (6) *"I met two interesting persons last night. The woman works for Clinton"*
- Abduktion: die Frau ist eine der beiden interessanten Personen und die andere Person ist ein Mann.
- $\{\text{person}(p_1), \text{person}(p_2), \text{man}(p_1), \text{woman}(p_2), \text{met}(I, p_1), \text{met}(I, p_2), \text{work-for}(p_2, \text{cl})\}$



4. Zusammenfassung

- Ein System kann mit Hilfe von einem Modellerzeuger die Bedeutung eines Textabschnittes erfassen
 - Die Eigenschaften der erzeugten Modelle (Minimalität, flache Struktur)
 - die hohe Ausdruckstärke der Logiksprache (λ -Typ-Sprache)
 - die Fähigkeit viele sprachliche Phänomene zu übersetzen
- gewährleisten zudem eine sehr hohe Genauigkeit dieser Bedeutung.



Literaturverzeichnis

- Gardent, C. and K. Konrad, *Interpreting definites using model generation*, Journal of Language and Computation 1 (2000), pp. 193-209
- Johan Bos, *Exploring Model Building for Natural Language Understanding*