

FACULTEIT INGENIEURSWETENSCHAPPEN

Master Computerwetenschappen

Masterproef *Jens Claes*

Promotor *Marc Denecker*

Begeleiders

Bart Bogaerts

Laurent Janssens

Academiejaar 2016-2017

P. Blackburn and J. Bos. Representation and inference for natural language. A first course in computational semantics. CSLI, 2005.

P. Blackburn and J. Bos. Working with discourse representation theory. An Advanced Course in Computational Semantics, 2006.



Automatisch vertalen van logigrammen naar logica

Motivatie

In het *Knowledge Base*-paradigma wordt een probleem gereduceerd tot een specificatie waarop verschillende inferenties worden uitgevoerd. Een formele specificatie is echter moeilijk te lezen en te schrijven. Het automatisch vertalen van natuurlijke taal naar logica lost dit probleem op.

Deze thesis onderzoekt het vertalen van een logigram (een puzzel met een aantal constraints in natuurlijke taal) naar een formele specificatie in logica

	9	Persoon			Kleur			Land					
	20	An	Bob	Charles	Dieter	Blauw	Groen	Rood	Paars	België	Frankrijk	Italië	Spanje
	Kat												
Dier	Hond				6 8				8 3				
ā	Paard				***								
	Varken						8 - 3		S 3				
	België											512	
р	Frankrijk												
Land	Italië									86 85			
	Spanje												
	Blauw						ne o	Ô	37	ř.			
ıı	Groen												
Kleur	Rood												
	Paars												

Voorbeeld - The contestant from Mount Union threw the black darts

Lexicon

Woord	Categorie	Betekenis		
the	Lidwoord (DET)	$\lambda R \cdot \lambda S \cdot \exists x \cdot R(x) \wedge S(x)$		
contestant	Substantief (NOUN)	$\lambda x \cdot contestant(x)$		
from	Voorzetsel (PREP)	$\lambda N \cdot \lambda P \cdot \lambda x \cdot N(\lambda y \cdot from(x, y) \wedge P(x))$		
Mount Union	Eigennaam (PN)	$\lambda P \cdot P(MountUnion)$		
threw	Overgankelijk werkwoord (TV)	$\lambda N1 \cdot \lambda N2 \cdot N2(\lambda x2 \cdot N1(\lambda x1 \cdot threw(x2, x1)))$		
the black darts	Eigennaam (PN)	$\lambda P \cdot P(TheBlackDarts)$		

Grammatica

Grammaticale regel	Betekenis
S → NP VP	$\llbracket VP \rrbracket (\llbracket NP \rrbracket)$
NP → DET N	$\llbracket DET \rrbracket (\llbracket N \rrbracket)$
NP → PN	$\llbracket PN rbracket$
N → NOUN NMOD	[NMOD]([NOUN])
NMOD → PREP NP	$\llbracket PREP \rrbracket (\llbracket NP \rrbracket)$
VP → TV NP	$\llbracket TV \rrbracket (\llbracket NP \rrbracket)$

$\boxed{[vp]([np])}$ $\exists x \cdot \text{from}(x, \text{MountUnion}) \land \text{contestant}(x) \land \text{threw}(x, \text{TheBlackDarts})$ $[\![\det]\!]([\![n]\!])$ [tv]([np]) $\lambda N2 \cdot N2(\lambda x2 \cdot threw(x2, TheBlackDarts))$ $\lambda S \cdot \exists x \cdot (from(x, MountUnion) \land contestant(x)) \land S(x)$ NP TVDET [nmod]([noun]) $\lambda R \cdot \lambda S \cdot \exists x \cdot R(x) \land S(x)$ $\lambda N1 \cdot \lambda N2 \cdot N2(\lambda x2 \cdot N1(\lambda x1 \cdot threw(x2, x1)))$ $\lambda x \cdot (from(x, MountUnion) \land contestant(x))$ $\lambda P \cdot P(TheBlackDarts)$ **NMOD NOUN** [prep]([np])the threw $\lambda P \cdot P(TheBlackDarts)$ $\lambda x \cdot contestant(x)$ $\lambda P \cdot \lambda x \cdot (from(x, MountUnion) \land P(x))$ NP **PREP** [pn]the black darts contestant $\lambda N \cdot \lambda P \cdot \lambda x \cdot N(\lambda y \cdot from(x, y) \wedge P(x))$ $\lambda P \cdot P(MountUnion)$

PN

 $\lambda P \cdot P(MountUnion)$

Mount Union

Een semantisch framework

(Blackburn en Bos 2005, 2006)

Het lexicon is verschillend per logigram. De grammatica is dezelfde voor alle logigrammen.

De betekenis van een woord is een functie van de lexicale categorie.

Compositionaliteit: de betekenis van een woordgroep is een combinatie van de betekenissen van de woorden waaruit ze bestaat. Zo wordt de betekenis van de woorden naar boven toe gepropageerd in de parse tree

Types en het formeel vocabularium

from

Veronderstelling: **elk woord** heeft **1 type** per logigram. Bij meerdere constraints **unificeren** de woorden die meerdere keren voorkomen de types. Verdere unificatie verloopt via vragen aan de gebruiker i.v.m. synonymie van woorden.

Substantieven en eigennamen introduceren een *basistype* Overgankelijke werkwoorden en voorzetsels introduceren een afgeleid *tuple-type* (met 2 basistypes als argument).

Eigennamen worden vertaald naar constanten van constructed types. Door unificatie van de basistypes worden deze eigennamen gegroepeerd.

Voorzetsels en overgankelijke werkwoorden introduceren een predicaat.

Extra axioma's (toegevoegd aan de theorie, specifiek voor logigrammen)

- Linken van predicaten a.d.h.v. hun signatuur
 - Bv. Twee predicaten met dezelfde signatuur zijn gelijk
- Elk predicaat is een bijectie
- Symmetrie-brekende axioma's

Resultaten

Gegeven: Aantal (basis)types, de constraints (in het Engels) en het logigram-specifiek lexicon

Extra vragen aan de gebruiker:

- Unificatie types (op basis van synonymie van woorden)
- Domein voor numerieke types

Resultaat: Vocabularium + Theorie in IDP

Experiment: Grammatica op basis van 10 puzzels **Evaluatie**: 10 nieuwe puzzels allemaal vertaalbaar mits 84 (kleine) correcties aan de constraints in natuurlijke taal. **Conclusie**: Succes mits beperking op gebruikte grammatica

Probleem	Aantal	Voorbeeld
Ontbrekende woorden	6	John's trip will begin before Janice's trip
Overtollige woorden	7	Wolfenden was said to be haunted by
> 1 type voor 1 woord	25	The trip starts begins at 9 and starts at Kiev
The one	15	before the one tour starting
Herschrijving NP	18	The comet Parks discovered by Parks
Type conversie	3	John finished before the man acting as doctor
Andere	11	\$5.99 \$6