Hoe leer je een grammatica uit voorbeelden?

Een computationele benadering

Barend Beekhuizen

Universiteit Leiden

15 december 2012 @ 5^{e} CogLingDagen

- Marrs drie niveaus van beschrijving:

functie	Wat is de functie (in wiskundige zin) die een systeem berekent
algoritme	Hoe representeert en berekent het systeem die functie
implementatie	Hoe voert het systeem deze berekening uit

- Marrs drie niveaus van beschrijving:

functie	Wat is de functie (in wiskundige zin) die een systeem berekent
algoritme	Hoe representeert en berekent het systeem die functie
implementatie	Hoe voert het systeem deze berekening uit

- Onze functie: *Betekenis* $\mapsto$ *Vorm*
- Typische aanpak:
  functie $>$ representatie $>$ berekening
- Expliciet in argument van sterke vs. zwakke equivalentie in lerende systemen (Berwick et al. 2011)
- Het primaat van de statische representatie (als onderzoeksdoel): betekenissen, structuren, opslag

- Wachten tot het functie- en algoritmische representatieniveau volledig beschreven zijn?

- Wachten tot het functie- en algoritmische representatieniveau volledig beschreven zijn?
- Inzichten uit het algoritmische berekeningsniveau.
- De structuur en inhoud van de representatie hangen af van het verwerkingsalgoritme

inhoud de functie van *er* in PP + V_{fin} + *er* + $NP_{subject}$ (Grondelaers),
inhoud de functie van geslacht op lidwoorden (Futrell & Ramscar),
structuur emergente complexiteit door beperkt geheugen/onvolledige input (Kirby)

- Wachten tot het functie- en algoritmische representatieniveau volledig beschreven zijn?
- Inzichten uit het algoritmische berekeningsniveau.
- De structuur en inhoud van de representatie hangen af van het verwerkingsalgoritme

inhoud de functie van *er* in PP + V_{fin} + *er* + $NP_{subject}$ (Grondelaers),
inhoud de functie van geslacht op lidwoorden (Futrell & Ramscar),
structuur emergente complexiteit door beperkt geheugen/onvolledige input (Kirby)

- Goed gezelschap:
  - Darwins idee van natuurlijke selectie is een algoritmisch proces (Dennett);
  - De functie is niet zo interessant ($Genenpoel^t \mapsto Genenpoel^{t+1}$), noch de representatie ervan.

- Mijn probleem: Leren van structuren die het mogelijk maken meerwoordsuitingen te maken.
- Blik op de representatie: welke structuren heb je nodig?
- Blik op het proces:
  - Hoe kom je aan die structuren?
  - Hoe bepaal je welke van de vele mogelijkheden de beste zijn?
- Wat leert die tweede blik ons?
  - De aard van die processen (bv. domeinalgemeenheid)
  - Waar een leerder naar 'zoekt' in taal als deze leert (compacte representaties, gemakkelijk vindbare patronen, expressiviteit, conservativiteit)
  - Hoe dat zoeken de structuur en inhoud van de representaties bepaalt.

- Centraal: Hoe verwerkt een leerder data zdd er representaties ontstaan waarmee ongeziene uitingen geproduceerd en verwerkt kunnen worden?
- Relevante deelvragen:
  - Hoe zoekt de leerder naar die representaties in de data?
  - Hoe evalueert de leerder de verschillende kandidaatrepresentaties?
  - Hoe gebruikt de leerder de representaties vervolgens?
- Wat zijn de representaties? Ook interessant, maar m.i. secundair.
- Verklaringsdoel is gedrag; verklaringsmiddel is cognitie

- In cognitief-taalkundige literatuur over kindertaalverwerving:
- Langacker (2009). 'A dynamic view of usage and language acquisition' *Cognitive Linguistics*

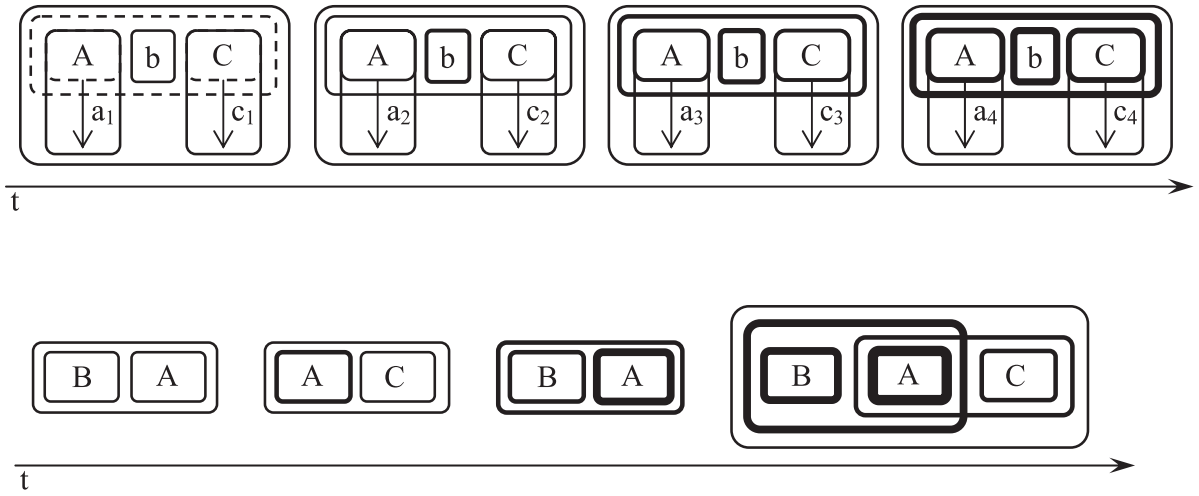


Figure: Twee illustraties van schemaverwerving (fig. 6 en 7 in Langacker 2009)

- In cognitief-taalkundige literatuur over kindertaalverwerving:
- Langacker (2009). 'A dynamic view of usage and language acquisition' *Cognitive Linguistics*

- Andere modellen:
  - Chang (2008): *Constructing Grammar: A computational model of the emergence of early constructions*.
  - Alishahi & Stevenson (2008): 'A Computational Model of Early Argument Structure Acquisition'. *Cognitive Science*.

- Computermodel: formele beschrijving van processen, structuren en inhouden die door een computer verwerkt kan worden
- Geen ruimte voor 'handwaving'
- Abstractie is onontkomelijk
- Mogelijkheid meer te doen dan 'met de hand'

- Wat wil een leerder?
  - Expressiviteit E
  - Conventionaliteit C
- Representaties moeten beide toelaten

- Wat wil een leerder?
  - Expressiviteit E
  - Conventionaliteit C
- Representaties moeten beide toelaten
- Maar soms zijn E en C in conflict:
  - E komt tot stand door abstractie
  - Maar abstractie verkleint C: meer 'moeite' om die specifieke representatie te maken.
  - Neem de data ab, ac en ad aan.
    - Grammatica 1: ab_S, ac_S, ad_S
    - Grammatica 2: aX_S, b_X, c_X, d_X
    - Grammatica 3: Yb_S, Yc_S, Yd_S, a_Y
    - Grammatica 4: YX_S, a_Y, b_X, c_X, d_X
  - Welke grammatica is 'optimaal'?

- Optimale representaties, hoe vind je die?
- Als je ze allemaal wil afgaan: niet
- Bounded Rationality (Gigerenzer): je werkt met heuristieken die je naar een lokaal optimum leiden met de minste moeite.
- Alternatief: rijkere universele grammatica
- Zoeken naar een grammatica binnen schier oneindige ruimte (dus: Marrs algoritmisch verwerkingsniveau)
- Hoe? Data + Analogie!

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):
  data Leerder heeft een corpus van uitingen gehoord;

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):
  - data Leerder heeft een corpus van uitingen gehoord;
  - G_0 Laat de aanvankelijke grammatica G_0 voor elk van die uitingen een productieregel bevatten (bv. $S \rightarrow abc$, $S \rightarrow ebc$);

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):
  - data Leerder heeft een corpus van uitingen gehoord;
  - G_0 Laat de aanvankelijke grammatica G_0 voor elk van die uitingen een productieregel bevatten (bv. $S \rightarrow abc$, $S \rightarrow ebc$);
  - zoek Vanuit G_t zoekt de leerder naar de beste alternatieve grammatica G' die één stap weg is van G_t;

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):
  - **data** Leerder heeft een corpus van uitingen gehoord;
  - **G_0** Laat de aanvankelijke grammatica G_0 voor elk van die uitingen een productieregel bevatten (bv. $S \rightarrow abc$, $S \rightarrow ebc$);
  - **zoek** Vanuit G_t zoekt de leerder naar de beste alternatieve grammatica G' die één stap weg is van G_t;
  - **stap** Voor elk paar van regels (r_i, r_j) die deel uitmaken van G_t:
    - kijk of ze koppelbare overlap hebben;
    - zo ja: maak een nieuwe regel van de overlap, en nieuwe regels van de verschillen en beschouw de vervanging van de oude door de nieuwe regels als een kandidaatgrammatica G'
    - dus $r_i = S \rightarrow abc$ en $r_j = S \rightarrow ebc$ levert $S \rightarrow Xbc$, $X \rightarrow a$ en $X \rightarrow e$ op
    - evalueer of die grammatica 'beter' is dan de huidige beste grammatica

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):

  data Leerder heeft een corpus van uitingen gehoord;

  G_0 Laat de aanvankelijke grammatica G_0 voor elk van die uitingen een productieregel bevatten (bv. $S \rightarrow abc$, $S \rightarrow ebc$);

  zoek Vanuit G_t zoekt de leerder naar de beste alternatieve grammatica G' die één stap weg is van G_t;

  stap Voor elk paar van regels (r_i, r_j) die deel uitmaken van G_t:

  - kijk of ze koppelbare overlap hebben;
  - zo ja: maak een nieuwe regel van de overlap, en nieuwe regels van de verschillen en beschouw de vervanging van de oude door de nieuwe regels als een kandidaatgrammatica G'
  - dus $r_i = S \rightarrow abc$ en $r_j = S \rightarrow ebc$ levert $S \rightarrow Xbc$, $X \rightarrow a$ en $X \rightarrow e$ op
  - evalueer of die grammatica 'beter' is dan de huidige beste grammatica

  naar G_{t+1} Kies de beste kandidaatgrammatica en laat die G_{t+1} zijn.

- BMM (Stolcke); aanpassing BMM* (Beekhuizen et al.):

  data Leerder heeft een corpus van uitingen gehoord;

  G_0 Laat de aanvankelijke grammatica G_0 voor elk van die uitingen een productieregel bevatten (bv. $S \rightarrow abc$, $S \rightarrow ebc$);

  zoek Vanuit G_t zoekt de leerder naar de beste alternatieve grammatica G' die één stap weg is van G_t;

  stap Voor elk paar van regels (r_i, r_j) die deel uitmaken van G_t:

  - kijk of ze koppelbare overlap hebben;
  - zo ja: maak een nieuwe regel van de overlap, en nieuwe regels van de verschillen en beschouw de vervanging van de oude door de nieuwe regels als een kandidaatgrammatica G'
  - dus $r_i = S \rightarrow abc$ en $r_j = S \rightarrow ebc$ levert $S \rightarrow Xbc$, $X \rightarrow a$ en $X \rightarrow e$ op
  - evalueer of die grammatica 'beter' is dan de huidige beste grammatica

  naar G_{t+1} Kies de beste kandidaatgrammatica en laat die G_{t+1} zijn.

  herhaal Herhaal **zoek**, **stap** en **naar $\mathbf{G_{t+1}}$** totdat de beste kandidaat-grammatica niet meer beter is dan G_t.

- Wat betekent 'beter'?
  - De beste balans tussen expressiviteit en dicht bij de data blijven (fit)
  - principe van Minimale Beschrijvingslengte van data en grammatica

- Wat betekent 'beter'?
  - De beste balans tussen expressiviteit en dicht bij de data blijven (fit)
  - principe van Minimale Beschrijvingslengte van data en grammatica
- Expressiviteit (beschrijvingslengte van de grammatica)
  - Hoe abstracter de grammatica, hoe meer er gegenereerd kan worden
  - De lengte van de grammatica geeft de abstractie weer
  - Hoe meet je die? Symbolen tellen en vermenigvuldigen met een constante per symbool

- Wat betekent 'beter'?
  - De beste balans tussen expressiviteit en dicht bij de data blijven (fit)
  - principe van Minimale Beschrijvingslengte van data en grammatica
- Expressiviteit (beschrijvingslengte van de grammatica)
- Fit met de data (beschrijvingslengte van de data)
  - Hoe concreter de grammatica, hoe dichter deze bij de data ligt
  - Stel dat elke regel een binaire identificatiecode heeft (bv. 100 of 110101)
  - Hoe langer die identificatiecode, hoe groter de cognitieve kosten
  - Dus: meer genestelde regels $\rightarrow$ kortere codes.
  - Codelengte van regel is logaritmische transformatie van relatieve frequentie van die regel
  - Codelengte van data is som van codelengte van alle gebruikte regels om het corpus te analyseren

- Wat betekent 'beter'?
  - De beste balans tussen expressiviteit en dicht bij de data blijven (fit)
  - principe van Minimale Beschrijvingslengte van data en grammatica
- Expressiviteit (beschrijvingslengte van de grammatica)
- Fit met de data (beschrijvingslengte van de data)
  - Hoe concreter de grammatica, hoe dichter deze bij de data ligt
  - Stel dat elke regel een binaire identificatiecode heeft (bv. 100 of 110101)
  - Hoe langer die identificatiecode, hoe groter de cognitieve kosten
  - Dus: meer genestelde regels $\rightarrow$ kortere codes.
  - Codelengte van regel is logaritmische transformatie van relatieve frequentie van die regel
  - Codelengte van data is som van codelengte van alle gebruikte regels om het corpus te analyseren
- Laagste som van data- en grammaticabeschrijvingslengte is de optimale grammatica

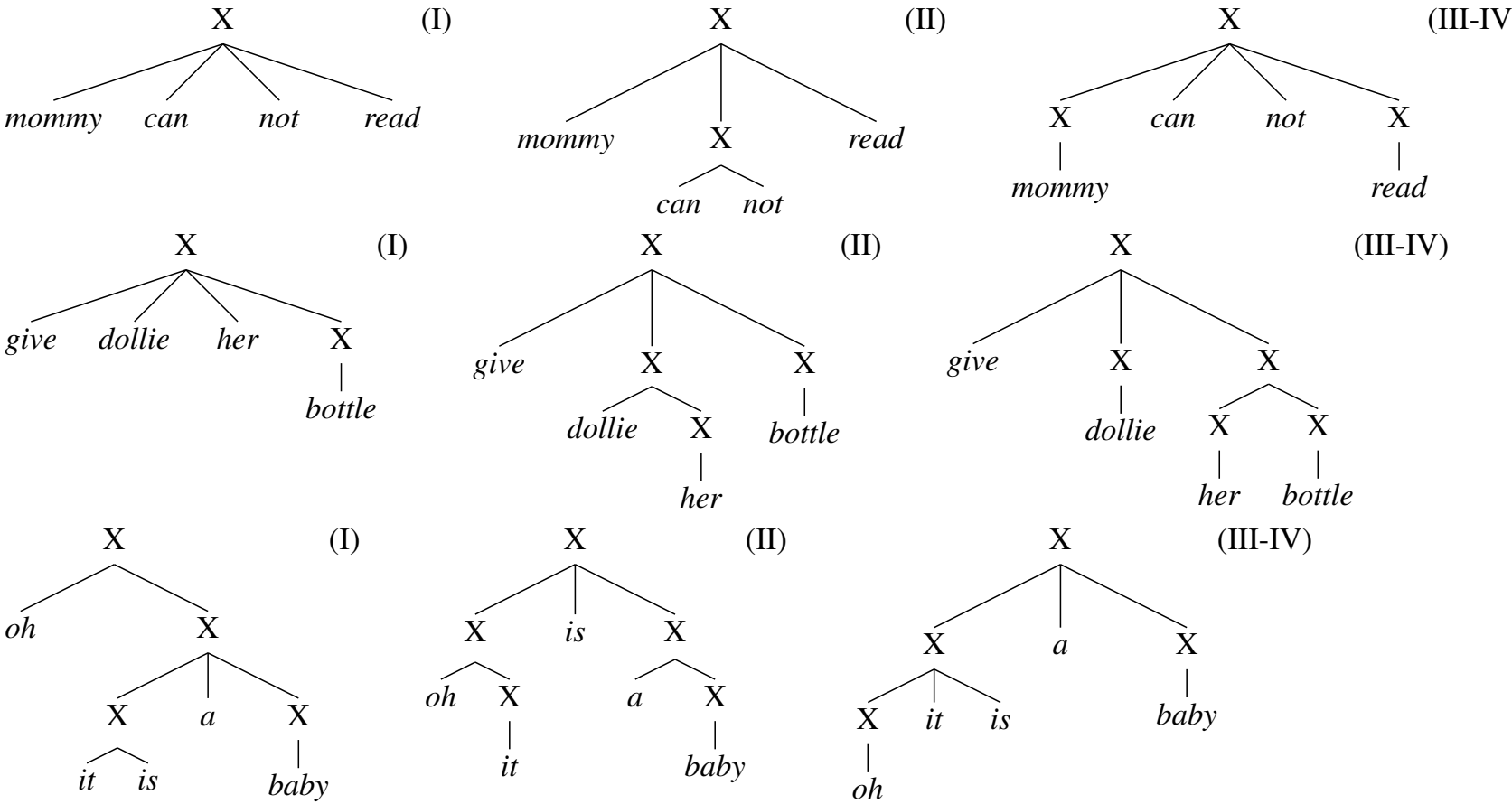
- Voorbeeld: *Ik wil brood*, *Ik wil een koekje*, *Brood is lekker*

- Voorbeeld: *Ik wil brood*, *Ik wil een koekje*, *Brood is lekker*
- G_0
  - G_0 is $[\textit{Ik wil brood}]_X$ (1), $[\textit{Ik wil een koekje}]_X$ (1), $[\textit{Brood is lekker}]_X$ (1)
  - Beschrijvingslengte (BL) grammatica: constante is $\log_2 8 = 4.0$: 10 symbolen $\times 4.0 = 40$
  - BL data: $-\log_2 \frac{1}{3} + -\log_2 \frac{1}{3} + -\log_2 \frac{1}{3} \approx 4.75$
  - Som $= 44.75$

- Voorbeeld: *Ik wil brood*, *Ik wil een koekje*, *Brood is lekker*
- G_0
- Kandidaatgrammatica, ronde 1
  - Stel: analogie tussen *Ik wil brood* en *Ik wil een koekje*
  - Dan: G' is $[\textit{Ik wil } X]_X$ (2), $[\textit{brood}]_X$ (1) $[\textit{een koekje}]_X$ (1), $[\textit{Brood is lekker}]_X$ (1)
  - BL grammatica: 9 symbolen $\times 4.0 = 36$
  - BL data: $(-\log_2 \frac{2}{4} + -\log_2 \frac{1}{4}) + (-\log_2 \frac{2}{4} + -\log_2 \frac{1}{4}) + (-\log_2 \frac{1}{4}) = 8$
  - $\mathrm{BL}(G_0) = 44.75$, $\mathrm{BL}(G') = 44$, dus G' is beter dan G_0: $G_1 = G'$

- Voorbeeld: *Ik wil brood*, *Ik wil een koekje*, *Brood is lekker*
- G_0
- Kandidaatgrammatica, ronde 1
- Kandidaatgrammatica, ronde 2
  - Stel: analogie tussen *brood* en *Brood is lekker* (ene is substring van andere)
  - Dan: G' is [*Ik wil X*]$_X$ (2), [*brood*]$_X$ (2) [*een koekje*]$_X$ (1), [*X is lekker*]$_X$ (1)
  - BL grammatica: 9 symbolen $\times \pm 4.0 = 36$
  - BL data: $(-\log_2 \frac{2}{5} + -\log_2 \frac{2}{5}) + (-\log_2 \frac{2}{5} + -\log_2 \frac{1}{5}) + (-\log_2 \frac{1}{5} + -log_2 \frac{2}{5}) = 9.93$
  - $\mathrm{BL}(G_1) = 44$, $\mathrm{BL}(G') = 45.93$, dus G' is niet beter dan G_1
  - Aangezien er verder geen analogieën meer te maken zijn en G' niet beter is dan G_1, is G_1 optimaal

- Toegepast op kindgerichte taal in Eve-deel van Brown corpus.
- Slechts één categorie: X en geen betekenis
- Getraind op 100, 500, 1000 of 2000 inputuitingen
- Voorbeelden



- Toegepast op kindgerichte taal in Eve-deel van Brown corpus.
- Slechts één categorie: X en geen betekenis
- Getraind op 100, 500, 1000 of 2000 inputuitingen
- Voorbeelden
- Borensztajn: grammatica achter geproduceerde uitingen van kind wordt abstracter en hierarchischer naarmate de leerder meer input krijgt
- Cf. gebruiksgebaseerde literatuur (Tomasello, Goldberg)
- Ook voor verwerking? Test op dezelfde honderd zinnen (voor vergelijkbaarheid):

n trainingitems	gem. *n* regels per uiting	gem. maximale inbedding van regels per uiting
100	2.25	1.90
500	2.80	2.18
1000	3.03	2.33
2000	3.53	2.55

- Doorzoeken van latente hypotheseruimte met simpele heuristiek: analogie over taalervaringen
- Daardoor: interessante verklaring op het algoritmisch berekeningsniveau
- Draagt bij aan inzichten over hoe we realistisch van geen naar een grammatica kunnen komen

- Doorzoeken van latente hypotheseruimte met simpele heuristiek: analogie over taalervaringen
- Daardoor: interessante verklaring op het algoritmisch berekeningsniveau
- Draagt bij aan inzichten over hoe we realistisch van geen naar een grammatica kunnen komen
- Er blijven wel bezwaren:
  - geen (echte) incrementaliteit,
  - input-is-uptake-aanname
  - geen betekenis of categorieën
  - nog steeds veel rekenkracht nodig - realistisch?
  - rule-list fallacy (ab, ac, ad óf aX, b, c, d)

- Minder berekening, meer 'one-shot' leren (recent werk: Medina et al.)
- Een verschil tussen input en uptake (cf. MOSAIC; Freudenthal et al.)
- Geen zinsbrede templaten, maar kleiner beginnen
- Bv. opvallende brokstukjes (meestal: woorden) en de relaties/afhankelijkheden daartussen (cf. Pivot Grammar)
- ...

Dank u!