Stratified reference: measurement

Implementation of Champollion (2015)

Mehdi Ghanimifard

University of Gothenburg

Semantics and Pragmatics, 2016

Outline

Outline

The puzzle

Why does pseudopartitives reject certain measure functions?

- five liters of water (volume)
- *five degrees Celsius of water (*temperature)

Outline

Previous work

Monotonicity

Schwarzschild (2006): Only monotonic measure functions are admissible.

- A measure function μ is monotonic iff for any two entities a and b, if a is a proper part of b, then $\mu(a) < \mu(b)$.

Previous work

Monotonicity

Schwarzschild (2006): Only monotonic measure functions are admissible.

- A measure function μ is monotonic iff for any two entities a and b, if a is a proper part of b, then $\mu(a) < \mu(b)$.
- For example, volume is monotonic, but temperature is not monotonic.

Previous work: Problem?

Monotonicity

- Five feet of snow fell on Berlin
- The height for snow is not monotonic, otherwise the snow in West Berlin should have lower height compering to whole snow in Berlin.
- Schwarzschild response: *a* should be a proper "Pragmatic Part" of *b*.

Outline

Novel observation

for-adverbials

- The same measurement functions rejected by psudopartitives are rejected by for-adverbials.
- five hours of driving (duration)
- *five miles per hour of driving (*speed)

Strategy: Solve the aspect puzzle and the measurement puzzle in distributivity by introducing the stratified reference.

Outline

Measurement Puzzle
The puzzle
Previous work
Novel observation
Solution
For-adverbials
Stratified reference
Pseudopartitives
Implementation
Presupposition
For-adverbials
Lambda Calculator

For-adverbials

Stratified reference presupposition

- For-adverbials presuppose stratified reference.
- eat apples for three hours

$$\forall e[[\![eat\ apple]\!](e) \rightarrow e \in *\lambda e' \left(\begin{array}{l} [\![eat\ apple]\!](e') \wedge \\ \varepsilon(\lambda t[hour(t) = 3])(runtime(e')) \end{array} \right)]$$

- *eat ten apples for three hours

$$\forall e[[\![eat\ ten\ apples]\!](e) \rightarrow e \in *\lambda e' \left(\begin{array}{l} [\![eat\ ten\ apples]\!](e') \wedge \\ \varepsilon(\lambda t[hour(t) = 3])(runtime(e')) \end{array} \right)$$

(Every eating-ten-apples event consists of subevents of eating-ten-apples with runtime of shorter than 3 hours)

Outline

Stratified reference

Definition

- Definition:

  $$SR_{f,\varepsilon(K)}(P) = \forall x[P(x) \to x \in {}^{*}\lambda y(P(y) \wedge \varepsilon(K)(f(y)))]$$

  (Any x with property P consists of parts who have property P and these parts are granular in f dimension with the scale of K)
- $\varepsilon(K)(f(y))$ roughly means that the f-dimension of y is small in scale of K.
- Intuitively, $x \in *\lambda y.P(y)$ means that x consists of one or more parts such that P holds for every part.

Outline

Pseudopartitives

Parallels between for-adverbials and pseudopartitives

1. Both reject if it fails to apply to parts of entity or event.

$$SR_{f,\varepsilon(K)}(P) = \forall x[P(x) \rightarrow x \in {}^{*}\lambda y(P(y) \wedge \varepsilon(K)(f(y)))]$$

*five pounds of book. (with "book" as a count noun)

Pseudopartitives

Parallels between for-adverbials and pseudopartitives

1. Both reject if it fails to apply to parts of entity or event.

   $$SR_{f,\varepsilon(K)}(P) = \forall x[P(x) \rightarrow x \in {}^{*}\lambda y(P(y) \wedge \varepsilon(K)(f(y)))]$$

   *five pounds of book. (with "book" as a count noun)
2. Both reject if the value of measure function stays constant acorss the parts of entity or event.

   $$SR_{f,\varepsilon(K)}(P) = \forall x[P(x) \rightarrow x \in {}^{*}\lambda y(P(y) \wedge \varepsilon(K)(f(y)))]$$

   *five degrees Celsius of the water in the bottle.

Pseudopartitives

Snow example

- five feet snow

$$SR_{height,\varepsilon(\lambda l.[feet(l)=5])}(\lambda y.snow(y)) =$$

$$\forall x[snow(x) \rightarrow x \in$$
$$*\lambda y \left(\begin{array}{l} \text{snow(y)} \wedge \\ \varepsilon(\lambda t[feet(t) = 5])(height(y)) \end{array} \right)]$$

Pseudopartitives

Mass vs count nouns

- 100 grams of apple

$$SR_{weight,\varepsilon(\lambda l.[gram(l)=100])}(\lambda y.apple(y)) =$$

$$\forall x[apple(x) \rightarrow x \in *\lambda y \left(\begin{array}{l} \text{apple(y)} \wedge \\ \varepsilon(\lambda t[gram(t) = 100])(weight(y)) \end{array} \right)]$$

- *100 grams of apples

$$SR_{weight,\varepsilon(\lambda l.[gram(l)=100])}(\lambda y. * apple(y)) =$$

$$\forall x[*apple(x) \rightarrow x \in *\lambda y \left(\begin{array}{l} \text{*apple(y)} \wedge \\ \varepsilon(\lambda t[gram(t) = 100])(weight(y)) \end{array} \right)]$$

Outline

Presupposition

Partial function

Partial function are used to represent presuppositions. The representation of partial function in Champollion (2015):

$$\lambda x : \phi . \psi$$

If ϕ holds it returns ψ otherwise undefined.

- My solution:

$$\lambda x . \partial(\phi)(\psi)$$

In Lambda Calculator:

$$\lambda x . partial(phi)(psi)$$

Outline

For-adverbials

For-adverbials for entities and events

Based on Champollion (2015) *for* is represented as follows:

$$[\![for]\!] = \lambda\tau_{\langle v,i\rangle}\lambda M_{\langle i,t\rangle}\lambda P_{\langle v,t\rangle}\lambda e.\partial(SR_{\tau,\varepsilon(M)}(P))(P(e)\wedge M(\tau(e)))$$

For example: "he walked [for five miles]$_{AdvP}$" :

$$[\![for\ five\ miles]\!] = \lambda P_{\langle v,t\rangle}\lambda e.\partial(SR_{\sigma,\varepsilon(\lambda l.[mile(l)=5])}(P))(P(e)\wedge[mile(\sigma(e)) = 5]$$

(σ is the parameter for spatial extend, instead of runtime.)

$$[\![walk\ \ for\ \ five\ \ miles]\!] =$$

$$\lambda e.\partial(SR_{\sigma,\varepsilon(\lambda l.[mile(l)=5])}(\lambda e'.walk(e'))(walk(e) \wedge [mile(\sigma(e)) = 5])$$

For-adverbials

For-adverbials for entities and events

On entities, for example “five miles of railroad tracks”:

$$[\![(for)]\!] = \lambda f_{\langle e,i\rangle}\lambda K_{\langle i,t\rangle}\lambda P_{\langle x,t\rangle}\lambda x.\partial(SR_{f,\varepsilon(K)}(P))(P(x) \wedge K(f(x)))$$

$$\Rightarrow [\![(for)\ five\ miles]\!] =$$

$$\lambda P_{\langle e,t\rangle}\lambda x.\partial(SR_{\sigma,\varepsilon(\lambda l.[mile(l)=5])}(P))(P(x) \wedge [mile(\sigma(x)) = 5])$$

$$\Rightarrow [\![[(for)\ five\ miles]\ of\ railroad\ tracks]\!] =$$

$$\lambda x.\partial(SR_{\sigma,\varepsilon(\lambda l.[mile(l)=5])}(\lambda y.railroad(y)))(railroad(x) \wedge [mile(\sigma(x)) = 5])$$

Outline

Lambda Calculator

Implementation

- In our application, the partial function is only for truth-values:
  constants of type $< t, < t, t >>$: *partial*
- Granularity function:
  constants of type $< it, it >$: *eps*
- Stratified reference parametrized constant function can be used in presupposition:
  constants of type $< ei, < it, < et, t >>>$: *SR*
  $\Rightarrow SR(f_{ei})(\varepsilon(K_{it}))(P_{et})$
  constants of type $< vi, < it, < vt, t >>>$: *SRv*
  $\Rightarrow SRv(T_{vi})(\varepsilon(M_{it}))(Q_{vt})$

Thank you

Thank you