Numeral Semantics | Wednesday

Lisa Bylinina & Rick Nouwen

ESSLLI 2019

bit.ly/esslli-numsem

Numeral Semantics: so far

$[\![\text{twelve}]\!]$

Twelve students came to the party

Twelve students can fit in the lift

The **twelve** students on this list all passed

Two is a Fibonacci number

Numeral Semantics: so far

$[\![$twelve$]\!]$

Twelve students came to the party

Twelve students can fit in the lift

The **twelve** students on this list all passed

Two is a Fibonacci number

e entity	$\langle e, t\rangle$ $\lambda x.\#x = 12$	$\langle\langle e, t\rangle, t\rangle$ quantifier	$\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$ $\lambda A.\lambda B.\lvert A \cap B\rvert = 12$
d 12	$\langle d, t\rangle$ degree property	$\langle\langle d, t\rangle, t\rangle$ degree quantifier	

Numeral Semantics: so far

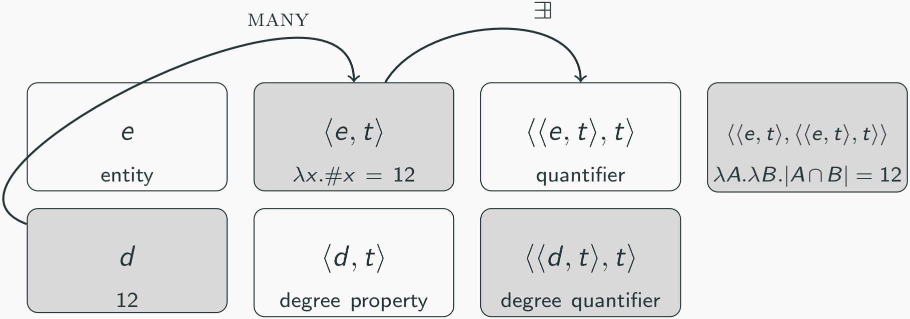
⟦twelve⟧

Twelve students came to the party

Twelve students can fit in the lift

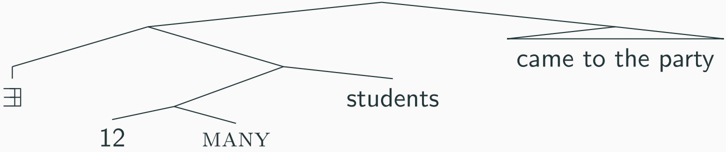
The **twelve** students on this list all passed

Two is a Fibonacci number



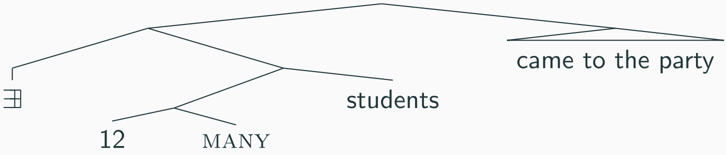
Numeral Semantics, so far: a prediction

$$\exists x[\#x = 12 \;\wedge\; {}^{*}\text{student}(x) \;\wedge\; {}^{*}\text{came-to-the-party}(x)]$$



Numeral Semantics, so far: a prediction

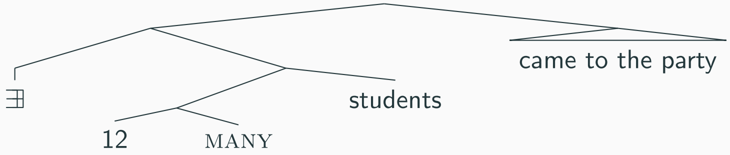
$$\exists x[\#x = 12 \ \wedge \ {}^{*}\text{student}(x) \ \wedge \ {}^{*}\text{came-to-the-party}(x)]$$



Crucial prediction: an *at least* (i.e. lower-bounded) reading

Numeral Semantics, so far: a prediction

$\exists x[\#x = 12 \;\wedge\; {}^{*}\text{student}(x) \;\wedge\; {}^{*}\text{came-to-the-party}(x)]$



Crucial prediction: an *at least* (i.e. lower-bounded) reading

$$\exists x[\#x = 12 \wedge^{*} \text{student}(x) \wedge^{*} \text{came-to-the-party}(x)]$$
$$\Leftarrow \exists x[\#x = 13 \wedge^{*} \text{student}(x) \wedge^{*} \text{came-to-the-party}(x)]$$

Side-note: distributivity and collectivity

$[\![$Twelve students came to the party$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$\Leftarrow \exists x[\#x = 13 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

Side-note: distributivity and collectivity

$[\![$Twelve students came to the party$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$\quad \Leftarrow \exists x[\#x = 13 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$[\![$Twelve students lifted the piano together$]\!] =$

Side-note: distributivity and collectivity

$[\![$Twelve students came to the party$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$\quad \Leftarrow \exists x[\#x = 13 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$[\![$Twelve students lifted the piano together$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{lifted-the-piano-together}(x)]$

Side-note: distributivity and collectivity

$[\![$Twelve students came to the party$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$\quad \Leftarrow \exists x[\#x = 13 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{came-to-the-party}(x)]$

$[\![$Twelve students lifted the piano together$]\!] =$

$\exists x[\#x = 12 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{lifted-the-piano-together}(x)]$

$\quad \not\Leftarrow \exists x[\#x = 13 \wedge {}^{*}\text{student}(x) \wedge {}^{*}\text{lifted-the-piano-together}(x)]$

At least versus exactly

Q: Did John take ten biscuits?

A: Yes, he took eleven.

A: No, he took eleven.

At least versus exactly

Q: Did John take ten biscuits?

A: Yes, he took eleven.

A: No, he took eleven.

Everyone who answered 10 questions correctly passes atleast

At least versus exactly

Q: Did John take ten biscuits?

A: Yes, he took eleven.

A: No, he took eleven.

Everyone who answered 10 questions correctly passes atleast

Everyone who answered 10 questions correctly fails exactly

A prominent traditional view

- Scalar implicature:

  if S entails S' while S' does not entail S,

  then uttering S' **implicates** that S is false
- Example:
  - All of the dots are blue

    entails Some of the dots are blue
  - Some of the dots are blue

    implicates Not all of the dots are blue

A prominent traditional view

- Scalar implicature:

  if S entails S' while S' does not entail S,

  then uttering S' **implicates** that S is false
- Example:
  - All of the dots are blue

    entails Some of the dots are blue
  - Some of the dots are blue

    implicates Not all of the dots are blue

- Similarly:
  - Thirteen people came to my party

    entails Twelve people came to my party
  - Twelve people came to my party

    implicates Not more than twelve people came to my party

Arguments & counter-arguments for implicated upper bounds

1. Cancellation

Some of the students came to the party. In fact, all of them did.

Twelve students came to the party. In fact, more did.

Twelve / Some of the students came to the party. *In fact, none did.

Arguments & counter-arguments for implicated upper bounds

1. Cancellation

Some of the students came to the party. In fact, all of them did.

Twelve students came to the party. In fact, more did.

Twelve / Some of the students came to the party. *In fact, none did.

Counter-argument: cancellation could be ambiguity resolution

Every student read loves some book, but no book was read by every student.

This morning I shot an elephant in my pyjamas. How he got in my pyjamas, I don't know.

Arguments & counter-arguments for implicated upper bounds

2. Negation kills implicatures

The soup is warm. $\rightsquigarrow$ the soup isn't hot.

The soup isn't warm. = the soup is cold

He didn't get 50% of the votes. = he got fewer

Arguments & counter-arguments for implicated upper bounds

2. Negation kills implicatures

The soup is warm. $\rightsquigarrow$ the soup isn't hot.

The soup isn't warm. = the soup is cold

He didn't get 50% of the votes. = he got fewer

Counter-argument:
Negation does not always operate on a lower-bounded reading

I liked it $\rightsquigarrow$ I didn't absolutely love it

#Neither of us liked the movie – she hated it and I absolutely loved it.

Neither of us have three kids - she has two, I have one.

(Horn 1996)

Arguments & counter-arguments for implicated upper bounds

3. Entailment patterns

Three of my friends own a red hat $\Rightarrow$ Three of my friends own a hat.

Exactly three of my friends own a red hat

$\not\Rightarrow$ Exactly three of my friends own a hat.

Arguments & counter-arguments for implicated upper bounds

3. Entailment patterns

Three of my friends own a red hat $\Rightarrow$ Three of my friends own a hat.

Exactly three of my friends own a red hat

$\not\Rightarrow$ Exactly three of my friends own a hat.

Counter-argument:

This intuition is compatible with numeral ambiguity.

Can we find cases where our intuition is in line with an exactly reading?

Arguments & counter-arguments for implicated upper bounds

3. Entailment patterns

Three of my friends own a red hat $\Rightarrow$ Three of my friends own a hat.

Exactly three of my friends own a red hat
$\not\Rightarrow$ Exactly three of my friends own a hat.

Counter-argument:
This intuition is compatible with numeral ambiguity.
Can we find cases where our intuition is in line with an exactly reading?

22.371.234 people voted for X $\not\Rightarrow$ 22.371.234 people voted.

Arguments & counter-arguments for implicated upper bounds

4. Another counter-argument

Q: Did John eat ten biscuits?

A: Yes/No

Compare to:

Sue takes milk or sugar in her tea. not both

Q: Do you take milk or sugar in your tea?

A: Yes, I take sugar.

A: ??No, I take both.

Interim conclusion

- Prenominal cardinals give rise to ambiguity:
  exactly versus *at least* reading
- What is the relation between those readings?

Interim conclusion

- Prenominal cardinals give rise to ambiguity:
  exactly versus *at least* reading
- What is the relation between those readings?

- So far, prenominal cardinals with $\exists$ yields an *at least* reading
- How can the *exactly* reading be derived from the *at least*?
- Implicature? We've just argued against this
- A mechanism that is more embedded in the grammar

Exhaustivity

Exhaustivity operator that attaches to a propositional node

$$[\![\text{ [} \textsc{exh} \ S \text{] }]\!] = 1$$

iff

$[\![S]\!] = 1$ & for any stronger alternative S' to S: $[\![S']\!] = 0$

Exhaustivity

Exhaustivity operator that attaches to a propositional node

$$[\![\ [\ \textsc{exh}\ S\ ]\ ]\!] = 1$$

iff

$[\![S]\!] = 1$ & for any stronger alternative S' to S: $[\![S']\!] = 0$

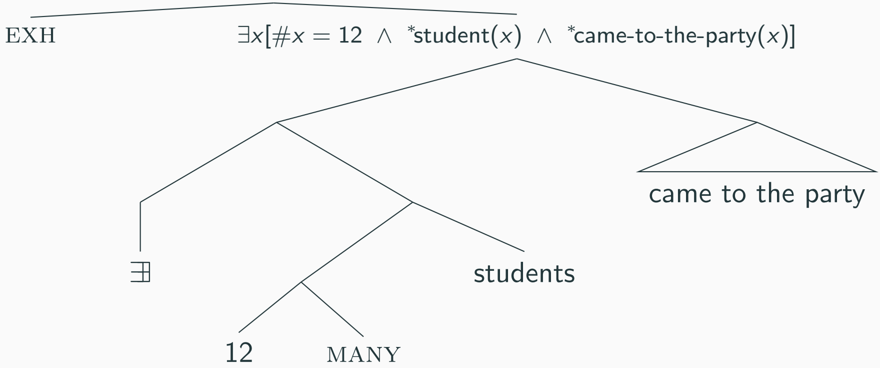
$[\![[\ \textsc{exh}$ [The soup is warm] $]]\!] = 1$

iff

The soup is warm & The soup is not hot

Exhaustivity

$\exists x[\#x = 12 \;\wedge\; {}^{*}\text{student}(x) \;\wedge\; {}^{*}\text{came-to-the-party}(x)] \wedge$
$\neg\exists x[\#x = 13 \;\wedge\; {}^{*}\text{student}(x) \;\wedge\; {}^{*}\text{came-to-the-party}(x)]$



Exhaustivity and scope

(1) You are allowed to tick two boxes $\Diamond > (\textsc{exh}) > \exists > 2$

Exhaustivity and scope

(1) You are allowed to tick two boxes $\Diamond > (\textsc{exh}) > \exists > 2$

(2) You are allowed to eat two biscuits $\textsc{exh} > \Diamond > \exists > 2$

Exhaustivity and scope

(1) You are allowed to tick two boxes $\Diamond > (\textsc{exh}) > \exists > 2$

(2) You are allowed to eat two biscuits $\textsc{exh} > \Diamond > \exists > 2$

(3) Some students answered three of the questions correctly

Exhaustivity and scope

(1) You are allowed to tick two boxes $\Diamond > (\textsc{exh}) > \exists > 2$

(2) You are allowed to eat two biscuits $\textsc{exh} > \Diamond > \exists > 2$

(3) Some students answered three of the questions correctly

Parallel to (2), we predict the following reading for (3):

$$\textsc{exh} > \text{some} > \exists > 3$$

no student answered more than three questions (not attested)

Constraints on scope of exhaustivity

Reminiscent of Heim 2000, the so-called *Heim/Kennedy generalisation*:

Nominal quantifiers intervene, where intensional quantifiers do not

Heim 2000, Nouwen & Dotlacil 2018

Constraints on scope of exhaustivity

Reminiscent of Heim 2000, the so-called *Heim/Kennedy generalisation*:

Nominal quantifiers intervene, where intensional quantifiers do not

Heim 2000, Nouwen & Dotlacil 2018

Heim introduces this as a constraint on degree expressions:

$$*\lambda d > \text{nominal quantifier} > d$$

Constraints on scope of exhaustivity

Reminiscent of Heim 2000, the so-called *Heim/Kennedy generalisation*:

Nominal quantifiers intervene, where intensional quantifiers do not

Heim 2000, Nouwen & Dotlacil 2018

Heim introduces this as a constraint on degree expressions:

$$*\lambda d > \text{nominal quantifier} > d$$

On the current proposal, it is not clear why we observe this constraint

(3) Some students answered three of the questions correctly

[exh [$_t$ [$_{ett}$ some students] [answered [$_{ett}$ $\exists$ [$_{et}$ [$_{et}$ three$_d$ many] students]]]]]

Numerals as degree quantifiers

e entity	$\langle e, t\rangle$ $\lambda x.\#x = 12$	$\langle\langle e, t\rangle, t\rangle$ quantifier	$\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$ $\lambda A.\lambda B.\lvert A \cap B\rvert = 12$
d 12	$\langle d, t\rangle$ degree property	$\langle\langle d, t\rangle, t\rangle$ degree quantifier	

Numerals as degree quantifiers

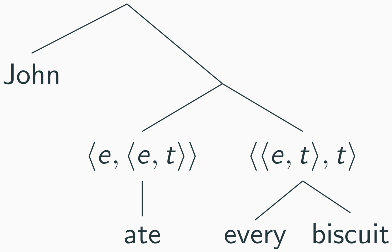
e entity	$\langle e, t\rangle$ $\lambda x.\#x = 12$	$\langle\langle e, t\rangle, t\rangle$ quantifier	$\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$ $\lambda A.\lambda B.\|A \cap B\| = 12$
d 12	$\langle d, t\rangle$ degree property	$\langle\langle d, t\rangle, t\rangle$ degree quantifier	

$[\![\text{twelve}]\!] = \lambda D.max(D) = 12$ (Kennedy 2015)

Numerals as type $\langle 1\rangle$ generalized quantifiers

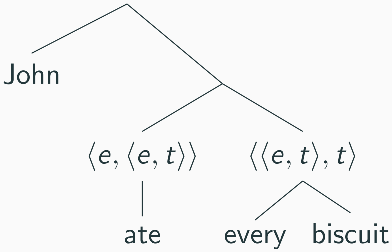
Type clashes and movement

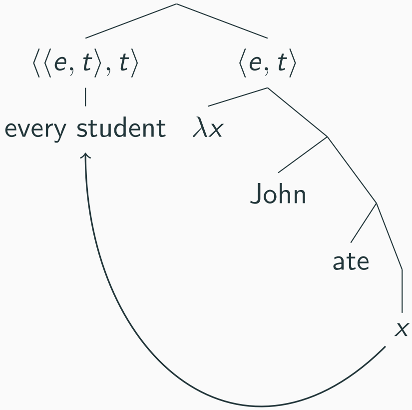
Very general assumption: type clashes are resolved by scope taking



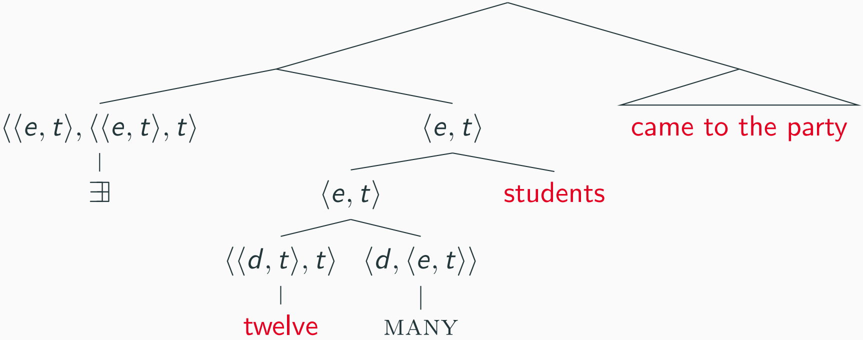
Type clashes and movement

Very general assumption: type clashes are resolved by scope taking

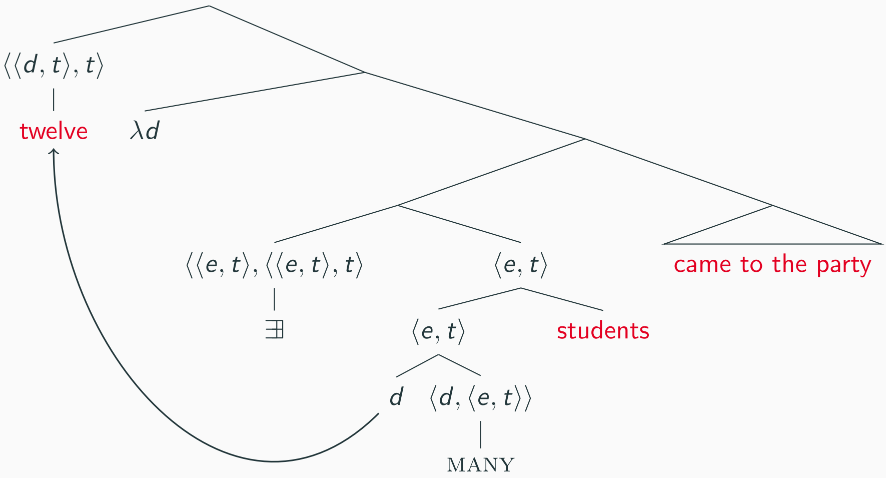




Type clash and movement



Type clash and movement



The degree quantifier view (Kennedy 2015)

$[\![\text{twelve}]\!] \ = \ \lambda D.max(D) = 12$

$[\![\lambda d.\ \exists\ d\text{-}\textsc{many}\ \text{students came to the party}]\!] \ =$
$\{(0,)1, 2, 3, 4, \ldots, k\}$

where $k =$ the number of students that came to the party

The degree quantifier view (Kennedy 2015)

$[\![\text{twelve}]\!] = \lambda D.max(D) = 12$

$[\![\lambda d.\ \exists\ d\text{-MANY students came to the party}]\!] =$
$\{(0,)1,2,3,4,\ldots,k\}$
where $k =$ the number of students that came to the party

$[\![\text{Twelve students came to the party}]\!] = 1$
iff $max(\{(0,)1,2,3,4,\ldots,k\}) = 12$

The degree quantifier view (Kennedy 2015)

$[\![\text{twelve}]\!] \;=\; \lambda D.max(D) = 12$

$[\![\lambda d.\ \exists\ d\text{-MANY students came to the party}]\!] \;=$
$\{(0,)1, 2, 3, 4, \ldots, k\}$
where $k =$ the number of students that came to the party

$[\![\text{Twelve students came to the party}]\!] = 1$
iff $max(\{(0,)1, 2, 3, 4, \ldots, k\}) = 12$
iff $k = 12$

Modals

You are allowed to tick two boxes $\Diamond > 2 > \exists$

it's allowed that the maximum number of boxes you tick is 2

= it's fine to tick exactly two boxes

You are allowed to eat two biscuits $2 > \Diamond > \exists$

the maximum number of biscuits you are allowed to eat is 2

Modals

You are allowed to tick two boxes $\Diamond > 2 > \exists$

it's allowed that the maximum number of boxes you tick is 2

$=$ it's fine to tick exactly two boxes

You are allowed to eat two biscuits $2 > \Diamond > \exists$

the maximum number of biscuits you are allowed to eat is 2

Some students answered three of the questions correctly

*2 $>$ some $> \exists$

Heim-Kennedy: *$\lambda d >$ some $> d$

The at least reading: type shifting (Partee 1986)

The at least reading: type shifting (Partee 1986)

$\text{BE} = \lambda Q.\lambda x.Q(\{x\})$ shift a quantifier to the set of entities such that the quantifier is true of each of the singleton sets formed by it

$\text{BE}(\lambda P.P(j)) = \lambda x.x = j$

$\text{IOTA} = \lambda P.\iota x.P(x)$

$\text{IOTA}(\text{BE}(\lambda P.P(j))) = j$

The at least reading: type shifting (Partee 1986)

$\textsc{be} = \lambda Q.\lambda x.Q(\{x\})$ shift a quantifier to the set of entities such that the quantifier is true of each of the singleton sets formed by it

$\textsc{be}(\lambda P.P(j)) = \lambda x.x = j$

$\textsc{iota} = \lambda P.\iota x.P(x)$

$\textsc{iota}(\textsc{be}(\lambda P.P(j))) = j$

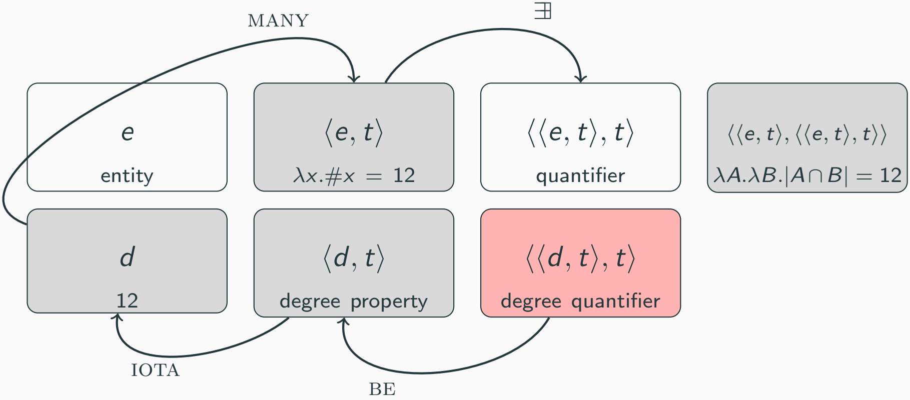
similarly,

$[\![\text{twelve}]\!] \ = \ \lambda D.max(D) = 12$

$\textsc{be}([\![\text{twelve}]\!])$ is the set of degrees that each interval in $[\![\text{twelve}]\!]$ shares, that is, $\{12\}$. So,

$\textsc{iota}(\textsc{be}([\![\text{twelve}]\!])) \ = \ 12$

Overview


MANY
∃
e
entity
$\langle e, t\rangle$
$\lambda x.\#x = 12$
$\langle\langle e, t\rangle, t\rangle$
quantifier
$\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$
$\lambda A.\lambda B.|A \cap B| = 12$
d
12
$\langle d, t\rangle$
degree property
$\langle\langle d, t\rangle, t\rangle$
degree quantifier
IOTA
BE

Interim conclusion

- Kennedy packages maximality and scope together
- It's a clear benefit for scope-taking,
- But maybe inherent maximality is not a virtue?
- One argument may come from zero

Zero in the degree quantifier framework

$$[\![\text{zero}]\!] = \lambda P.max(P) = 0$$

Zero in the degree quantifier framework

$$[\![\text{zero}]\!] = \lambda P.max(P) = 0$$

$$[\![\text{I have zero emails in my inbox}]\!]$$

$$=$$

$$max(\lambda d.\text{I have } d\text{-}\textsc{many} \text{ emails in my inbox}) = 0$$

$$=$$

there are exactly zero emails in my inbox

Zero in the degree quantifier framework

$$[\![\text{zero}]\!] = \lambda P.max(P) = 0$$

$$[\![\text{I have zero emails in my inbox}]\!]$$

$$=$$

$$max(\lambda d.\text{I have } d\text{-}\textsc{many} \text{ emails in my inbox}) = 0$$

$$=$$

there are exactly zero emails in my inbox

Note, then, that zero is predicted to mean the same as no.

Zero versus no

No students have read my book, have they / *haven't they?

Zero people love her, *do they / don't they? (DeClercq 2011)

Zero versus no

No students have read my book, have they / *haven't they?

Zero people love her, *do they / don't they? (DeClercq 2011)

No students have visited me in years.

*Zero students have visited me in years.

(Zeijlstra 2007, Gajewski 2011, Bylinina & Nouwen 2018)

The degree quantifier analysis wrongly predicts that zero licenses NPIs.

Severing maximality from scope-taking

Blok, Bylinina, Nouwen 2018, cf. Buccola 2017)

$[\![\text{twelve}]\!] = 12$

$[\![\textsc{Quant}]\!] = \lambda n.\lambda P.P(n)$

$[\![\textsc{Quant}\ \text{twelve}]\!] = \lambda P.P(12)$

$[\![\textsc{Max}]\!] = \lambda D_{\langle\langle d,t\rangle,t\rangle}.\lambda P.max(P) \in \cap D.$

$[\![\textsc{Max}\ [\ \textsc{Quant}\ \text{twelve}\ ]]\!] = \lambda P.max(P) \in \{12\}$

Severing maximality from scope-taking

Blok, Bylinina, Nouwen 2018, cf. Buccola 2017)

$[\![\text{twelve}]\!] = 12$

$[\![\textsc{Quant}]\!] = \lambda n.\lambda P.P(n)$

$[\![\textsc{Quant}\ \text{twelve}]\!] = \lambda P.P(12)$

$[\![\textsc{Max}]\!] = \lambda D_{\langle\langle d,t\rangle,t\rangle}.\lambda P.max(P) \in \cap D.$

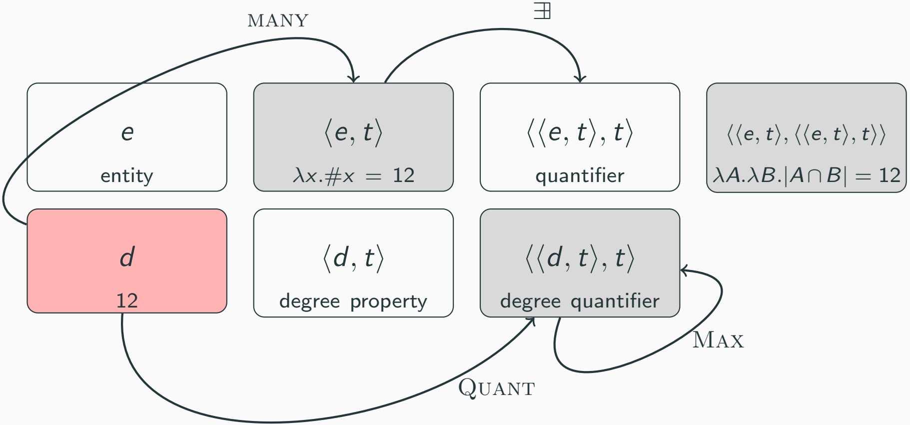
$[\![\textsc{Max}\ [\ \textsc{Quant}\ \text{twelve}\ ]]\!] = \lambda P.max(P) \in \{12\}$

You are allowed to eat two biscuits $\quad \textsc{Max} > (\textsc{Quant}\ \text{twelve}\ ) > \Diamond$

Some students answered three of the questions correctly

$*\textsc{Max} > (\textsc{Quant}\ \text{twelve}\ ) > \text{some}$

Overview



MANY
∃
e
entity
$\langle e, t \rangle$
$\lambda x.\#x = 12$
$\langle\langle e, t \rangle, t \rangle$
quantifier
$\langle\langle e, t \rangle, \langle\langle e, t \rangle, t \rangle\rangle$
$\lambda A.\lambda B.|A \cap B| = 12$
d
12
$\langle d, t \rangle$
degree property
$\langle\langle d, t \rangle, t \rangle$
degree quantifier
Max
Quant

NPIs and exhaustification

- The licensing of NPIs is sensitive to properties of the **non-exhaustified** meaning
- Gajewski's necessary condition for NPI licensing

> The NPI is in a non-trivially downward entailing environment, even if the exhaustifying operator were not there

Zero students have visited me in years

[(MAX) [QUANT zero]]] [λd [$\exists$ [d MANY students have visited me in years]]]

- On this final account, zero is **not** non-trivially downward entailing

See you tomorrow!

`bit.ly/esslli-numsem`