# Mirroring strong and weak NPIs and PPIs

Hedde Zeijlstra
Georg-August-Universität Göttingen
hzeijls@uni-goettingen.de

NayFest 2018

# I. Strong vs weak NPIs

- Negative Polarity Items (NPIs) differ with respect to the logical properties of their licensers.

- Weak NPIs are fine in all (Strawson-) Downward Entailing contexts.

- Strong NPIs are only fine in Anti-Additive contexts.

# I. Strong vs weak NPIs

- Weak NPIs: *any*, *ever*

*Mary bought any cookies

*Mary has ever been there

Nobody bought any cookies

Nobody has ever been there

# I. Strong vs weak NPIs

- Weak NPIs: *any*, *ever*

Few people bought any cookies

Few people have ever been there

At most 10 people bought any cookies

At most 10 people have ever been there

# I. Strong vs weak NPIs

- Strong NPIs: *in years*, punctual *until*

*Mary has been there in years

*Mary moved in until June

Nobody has been there in years

Nobody moved in until June

# I. Strong vs weak NPIs

- Strong NPIs: *in years*, punctual *until*

*Few people have been there in years

*Few people moved in until June

*At most 10 people have been there in years

*At most 10 people moved in until June

# II. Background 1

- Following Chierchia (2006, 2013), basing himself on Kadmon & Landman (1993), Krifka (1995) and Gajewski (2002), a sentence with an unlicensed NPI yields a logical contradiction and logical contradictions give rise to ungrammaticality judgments.
- The source of the logical contradiction is twofold:

  – NPIs introduce (sub)domain-alternatives;
  – NPIs come along with a syntactic feature that triggers the presence of a covert exhaustification operator.

# II. Background 1

*I have any potato

| | |
|---|---|
| [I have any potato$_{[u\sigma,D]}$] | no contradiction, unchecked feature |
| [EXH$_{[i\sigma,D]}$ I have any potato$_{[u\sigma,D]}$] | contradiction, checked feature |

I don't have any potato

[EXH$_{[i\sigma,D]}$ I don't have any potato$_{[u\sigma,D]}$ ]

# II. Background 1

*I have any potato:

$\exists p[p \in \{p1, p2, p3\}\ \&\ Have(I, p)] <$
$\exists p[p \in \{p1, p3\}\ \&\ Have(I, p)]$
$\exists p[p \in \{p2, p3\}\ \&\ Have(I, p)]$
$\exists p[p \in \{p1, p3\}\ \&\ Have(I, p)]$
$\exists p[p \in \{p1\}\ \&\ Have(I, p)]$
$\exists p[p \in \{p2\}\ \&\ Have(I, p)]$
$\exists p[p \in \{p3\}\ \&\ Have(I, p)]$

These domain alternatives are stronger. Therefore:

# II. Background 1

EXH($\exists$p[p$\in${p1, p2, p3} & Have(I, p)) =

$\exists$p[p$\in${p1, p2, p3} & Have(I, p)] &
$\neg\exists$p[p$\in${p1, p3} & Have(I, p]) &
$\neg\exists$p[p$\in${p2, p3} & Have(I, p)] &
$\neg\exists$p[p$\in${p1, p3} & Have(I, p)] &
$\neg\exists$p[p$\in${p1} & Have(I, p)] &
$\neg\exists$p[p$\in${p2} & Have(I, p)] &
$\neg\exists$p[p$\in${p3} & Have(I, p)]

- A clear contradiction

# II. Background 1

I don't have any potato

$\neg\exists p[p\in\{p1, p2, p3\}\ \&\ Have(I, p)] >$

$\neg\exists p[p\in\{p1, p2\}\ \&\ Have(I, p)]$

$\neg\exists p[p\in\{p2, p3\}\ \&\ Have(I, p)]$

$\neg\exists p[p\in\{p1, p3\}\ \&\ Have(I, p)]$

$\neg\exists p[p\in\{p1\}\ \&\ Have(I, p)$, etc.

- No domain alternative is stronger, so no contradiction arises.

$EXH(\neg\exists p[p\in\{p1, p2, p3\}\ \&\ Have(I, p)]) =$

$\neg\exists p[p\in\{p1, p2, p3\}\ \&\ Have(I, p)]$

# III. Background 2

- How under this approach does the strong-weak distinction follow?

- Gajewski (2011)/Chierchia (2013): Strong NPIs are not sensitive to AA contexts, but must be licensed by operators whose *enriched* meaning is still DE.

# III. Background 2

- *Few N* and *at most X N* infer an existential contribution:

Few students left ≈ Some, but not many students left

At most 10 students left ≈ At least one, but no more than 10 students left

# III. Background 2

- Enriched *Few N* and *at most X N* are no longer DE:

Some, but not many students left -/→
Some but not many students left in a hurry

At least one, but no more than 10 students left -/→
At least one, but no more than 10 students left in a hurry

# III. Background 2

- Enriched *Few N* and *at most X N* thus do not prevent NPIs from yielding a contradiction.

- AA elements (*nobody*, *never*) are never enriched in such a way that their DE property is disrupted.

# III. Background 2

- Strong and weak NPIs involve different kinds of exhaustifiers:
- Weak NPIs trigger the presence of an exhaustifier (EXH) that only evaluates the assertion with respect to its alternatives;
- Strong NPIs require the presence of an exhaustifier (EXH+) that evaluates all meaning levels (presupposition, assertion and implicatures) with respect to its alternatives.

# IV. Problem

- It is a property of a particular NPI whether it is weak or not, but under this analysis NPI-strength results from the interaction between the exhaustifier and the licenser.

- How can it be encoded on an NPI that its exhaustifier must consider the enriched meaning of the *licensing* element or not?

# IV. Problem

- One way would be imposing different features (say [uEXH] vs [uEXH+]), cf. Chierchia 2013 for a proposal along such lines.

- But such a solution faces two problems:

# IV. Problem

- It does not explain why one type of NPIs would carry [uEXH] and another type [uEXH+].

- Feature checking in general is a problem for Chierchia‘s approach, since NPIs can be licensed outside their syntactic domain.

I don‘t work in order to make any money
Nobody says she has ever been there

# IV. Problem

- Weakening the locality conditions on such Agree relations does not form a solution either, as other NPIs do obey syntactic locality conditions

*Nobody claims she has moved in until June

*Nobody claims she has stayed there in weeks

# V. Strict vs non-strict NPIs

- Two other types of NPIs:

- Strict NPIs: NPIs that have to be in the same syntactic domain as their licensers;

- Non-strict NPIs: NPIs that do not have to be in the same syntactic domain as their licensers.

# V. Strict vs non-strict NPIs

- Collins & Postal (2014), using another framework and different terms, make the observation that strong and weak NPIs behave differently with respect to syntactic locality:

  Strong NPIs are Strict NPIs
  Weak NPIs are Non-strict NPIs

- Strict NPIs have to be in the same syntactic domain as their licensers; non-strict NPIs don‘t.

# V. Strict vs non-strict NPIs

- Weak, non-strict NPIs:

I don‘t work in order to make any money

I don‘t travel to Moscow in order to ever have been in Russia

Nobody claims she has bought any books

Nobody claims she has ever been there

# V. Strict vs non-strict NPIs

- Strong, strict NPIs:

*I don‘t stay here in order to move in until June
*I don‘t stay that long in Moscow in order to have been there in weeks

*Nobody claims she has moved in until June
*Nobody claims she has stayed there in weeks

# VI. Syntactic exhaustification

- What the facts suggest is that only the relation between a strict NPI and its licenser involves syntactic Agree (i.e. feature checking).

- Only strict (and, therefore, strong) NPIs come along with some syntactic feature that triggers the presence of a covert exhaustifier.

# VII. Pragmatic exhaustification

- But what about non-strict NPIs?

- Syntactic Agree cannot underlie the relation between EXH and a non-strict NPI.

# VII. Pragmatic exhaustification

- **Proposal:** allude to a pragmatic mechanism that states that if some alternatives have been introduced in the sentence and they have not been applied to by any other operator that applies to alternatives, as a last resort, the entire clause is exhaustified. This proposal essentially treats NPIs along the lines of Krifka (1995).

- This is analogous to introduced scalar alternatives that have not been applied to by a focus-sensitive operator and receive an exhaustified interpretation.

# VII. Pragmatic exhaustification

- Weak/non-strict NPIs only introduce domain-alternatives.

- Since other, known focus-sensitive operators do not apply to domain alternatives (but to rather scalar or other kinds of alternatives), introduced domain alternatives must in principle be exhaustified at a pragmatic level.

# VII. Pragmatic exhaustification

- Strict NPIs are thus, in the most literal way, the grammaticalized versions of non-strict NPIs: the requirement that NPIs be exhaustified gets encoded by means of a syntactic feature.

– *Ever, any*: introduce domain alternatives

– *In weeks,* punctual *until*: introduce domain alternatives and carry [u$\sigma$,D].

# VII. Pragmatic exhaustification

- An advantage of this classification is that NPI-hood no longer depends on two (independent) properties, but only on one.

- But how to account, then, for the difference between strong and weak NPIs?

# VIII. Hypothesis

- Recall: strict NPIs are strong NPIs; non-strict NPIs are weak NPIs.

- The hypothesis, then, suggests itself:

# VIII. Hypothesis

- EXH+: an exhaustifier that can check off (by carrying [iσ,D]) a feature on an element carrying [uσ,D].

- EXH: an exhaustifier that can only be introduced as a last resort if a grammatical sentence (i.e. a sentence with no unchecked features) still contains unaffected domain alternatives.

# VIII. Hypothesis

- Syntactic exhaustification:
- is triggered by Agree;
- is subject to syntactic locality constraints;
- may apply at any position in the clause, provided its complement is of the right semantic type and it c-commands the PI;
- Involves EXH+.

- Pragmatic exhaustification:
- takes place as a last resort operation;
- is not subject to syntactic locality constraints;
- may apply at the CP level only (given that it is a last resort operation applying at clausal level);
- Involves EXH.

# IX. Question

- This hypothesis readily accounts for the fact that weak and strong NPIs trigger the presence of different exhaustifiers (the encoding problem).

- Is there any further empirical evidence for the difference between syntactic and pragmatic exhaustification?

# X. Universal Quantifier PPIs

- In principle, Chierchia's approach must also be applicable to universals, as nothing would rule out the introduction of domain alternatives in the restrictive clause of a universal quantifier.

- However, since universals are at the other end of the scale, the reasoning in terms of arising contradictions is reverse: such universal quantifiers that are obligatorily exhaustified are expected to be PPIs.

# X. Universal Quantifier PPIs

- Examples of such universal quantifier PPIs can be found in the domain of (deontic) modals, although such PPIs are not restricted to modals (cf. Iatridou & Zeijlstra 2013, Zeijlstra 2013, 2016, Homer 2015; see also Giannakidou & Mari 2016 for a different approach).

# X. Universal Quantifier PPIs

- Modal PPIs: in many languages certain (deontic) universal modals outscope negation, whereas other (deontic) universals modals scope under it. (Existential deontic modals always scope under negation.)

| | |
|---|---|
| John mustn‘t leave | Must > Neg |
| John shouldn‘t leave | Should > Neg |
| John doesn‘t have to leave | Neg > Have to |
| John can‘t leave | Neg > Can |

# X. Universal Quantifier PPIs

- Modals scoping over negation are PPIs.

- Iatridou & Zeijlstra (2013): every modal auxiliary reconstructs under negation, unless it is a PPI (if you can reconstruct, you must reconstruct).

John can‘t <can> leave

John doesn‘t have to leave

John mustn‘t <~~must~~> leave

# X. Universal Quantifier PPIs

- Evidence comes from the fact that in every case where PPIs may appear under negation, modals normally outscoping negation may also take scope under negation (cf. Homer 2015).

- Metalinguistic/contrastive negation
- Intervention effects
- Extra-clausal negation
- Baker-Szabolcsi-effects

# X. Universal Quantifier PPIs

- Metalinguistic/contrastive negation:

If you push the red button, you will see something, but if you press the blue button you WON'T see something.

A: One student {must/should} read 5 articles on the topic.

B: NO student {must/should} read 5 articles on the topic, but one student is {encouraged/allowed} to do so.

# X. Universal Quantifier PPIs

- Intervention effects:

John didn't offend someone because he was malicious (but because he was stupid).

John doesn't always call someone.

She {must/should} not marry him because he is handsome but because he is smart.

I {must/should} not always take the garbage outside. Many times my son {must/should} do that.

# X. Universal Quantifier PPIs

- Extra-clausal negation:

I didn't say that John called someone.

I regret that John called someone.

I won't say that John {must/should} leave.

I regret that John {must/should} leave.

# X. Universal Quantifier PPIs

- Baker-Szabolcsi-effects:

I am surprised that John didn't call someone.

Few boys didn't call someone.

I am surprised that he {must/#should} not write a paper about the Romans.

Very few doctors {must/#should} not work tonight; most of them are on duty.

# X. Universal Quantifier PPIs

- The PPI-hood of these modals follows directly, once they are assumed to be universal quantifiers that obligatorily introduce domain alternatives that must be exhaustified.

# X. Universal Quantifier PPIs

She mustn't leave (*NEG>MUST)

EXH(NOT(Must(leave(she))) =

$\text{EXH}[\neg\forall w[w \in \{w1, w2, w3\} \rightarrow \text{leave}_w(\text{she})]] =$

$\neg\forall w[w \in \{w1, w2, w3\} \rightarrow \text{leave}_w(\text{she})]$ &

$\neg\neg\forall w[w \in \{w1, w2\} \rightarrow \text{leave}_w(\text{she})]$ &

$\neg\neg\forall w[w \in \{w2, w3\} \rightarrow \text{leave}_w(\text{she})]$ &

$\neg\neg\forall w[w \in \{w1, w3\} \rightarrow \text{leave}_w(\text{she})]$ & …

Contradiction!

# X. Universal Quantifier PPIs

She mustn't leave (*NEG>MUST)

EXH(NOT(Must(leave(she))) =

$EXH[\neg\forall w[w \in \{w1, w2, w3\} \rightarrow leave_w(she)]] =$

$\neg\forall w[w \in \{w1, w2, w3\} \rightarrow leave_w(she)]$ &

$\forall w[w \in \{w1, w2\} \rightarrow leave_w(she)]$ &

$\forall w[w \in \{w2, w3\} \rightarrow leave_w(she)]$ &

$\forall w[w \in \{w1, w3\} \rightarrow leave_w(she)]$ & …

Contradiction!

# XI. PPI strenght

- Modal, universal PPIs also exhibit the strong-weak distinction. *Should* is a strong PPI that is banned from all DE contexts:

| | |
|---|---|
| Mary shouldn‘t leave | Should>neg |
| Mary should never leave | Should>never |
| Few students should leave | Should>few |
| At most 10 students should leave | Should>AM10 |

# XI. PPI strenght

- *Must,* by contrast, is a weak PPI that is banned only from AA contexts:

| | |
|---|---|
| Mary mustn‘t leave | Must>neg |
| Mary must never leave | Must>never |
| Few students must leave | Few>must |
| At most 10 students must leave | AM10>must |

# XI. PPI strenght

- This also explains the difference between *must* and *should* w.r.t. the Baker-Szabolcsi effects. Strong PPIs do not exhibit these effects.

{Weinig/*veel} mensen zijn {allerminst / inderdaad / verre van} van tevreden

Few people are not.in.the.least / indeed / far from happy

‘Few people aren‘t in the least happy‘

# XI. PPI strenght

- This also explains the difference between *must* and *should* w.r.t. the Baker-Szabolcsi effects. Strong PPIs do not exhibit these effects.

*Ik ben verbaasd dat je niet {allerminst/inderdaad /verre van} tevreden bent

I am surprised that you are not not.in.the.least / indeed / far from happy

‘I‘m surprised that few people aren‘t in the least happy‘

# XI. PPI strenght

- This distinction is not surprising. It is a well known fact that PPIs exhibit the same strong-weak distinctions as NPIs (cf. Van der Wouden 1994).

- A novel observation, however, is that weak universal quantifier PPIs exhibit linear-sensitive effects, which strong (universal quantifier ) PPIs don’t exhibit.

- Weak universal quantifier PPIs are only PPI-like if they precede their anti-licenser.

# XII. Linear-sensitivity

- Most speakers of English do not allow *must* to take scope below a negative indefinite object, but do allow it to take scope below a negative indefinite subject:

| | |
|---|---|
| Nobody must leave | Must>nobody;<br>Nobody>must |
| Mary must read nothing | Must>nothing;<br>*Nothing>must |

# XII. Linear-sensitivity

- However, *should* always takes scope above a negative indefinite

| | |
|---|---|
| Nobody should leave | Should>nobody |
| | *Nobody>should |
| Mary should read nothing | Should>nothing |
| | *Nothing>should |

# XII. Linear-sensitivity

- (Northern/western) Dutch *moeten* (‚must‘) is a weak PPI:

Marie moet niet vertrekken — Must>neg
Marie must not leave — *Neg>must

Hoogstens 3 mensen moeten vertrekken
At most 3 people must leave

AM3>must
Must>AM3

# XII. Linear-sensitivity

- But in subordinate clauses, *moeten*, being in sentence-final position (Dutch is OV and main clause-V2), lacks a PPI-effect:

... dat Marie niet moet vertrekken

... That Marie niet must leave

Must>neg

Neg>must

# XII. Linear-sensitivity

- Dutch and German *should*-verbs are strong PPIs:

| | |
|---|---|
| Hoogstens drie mensen zouden moeten vertrekken | |
| At most 3 people would must leave | Should>AM3 |
| ‘At most 3 people should leave’ | *AM3>should |
| | |
| Höchstens drei Leute sollen abfahren | Should>AM3 |
| At most 3 people should leave | *AM3>should |

# XII. Linear-sensitivity

- Also, these Dutch and German *should*-verbs don‘t show the linear-sensitivity effect:

… dat Jan niet zou moeten vertrekken

… that Jan neg would must leave

‘… that Jan shouldn’t leave’

Should >neg;*neg> should

… dass Hans nicht abfahren soll

… that Hans neg leave should

‘… that Hans shouldn’t leave’

Should >neg;*neg> should

# XII. Linear-sensitivity

*Must* and Dutch *moeten* are weak PPIs and may scope under negation when they surface below negation, but not when they surface above negation.

*Must* and Dutch *moeten* cannot reconstruct below negation.

*Should*, German *sollen* and Dutch *zouden moeten* are strong PPIs and cannot take immedeate scope below negation.

# XII. Linear-sensitivity

These facts are not restricted to modals. Dutch *ieder* (‚every‘) behaves similarly in this respect (and counts as being a weak PPI, cf. Zeijlstra 2016). Similar effects apply in Northern German varieties, Libanese Arabic and Japanese:

I didn‘t see everybody $\neg > \forall$

Ik heb niet iedereen gezien $\neg > \forall$

I have not everybody seen

# XII. Linear-sensitivity

Every boy didn't walk — English

OK: "No boy walked"

OK: "Not every boy walked"

Iedere jongen liep niet — Dutch

OK: "No boy walked"

*: "Not every boy walked"

# XII. Linear-sensitivity

Dutch *ieder* (‚every‘) is again a PPI that may scope under negation when it surfaces below negation, but not when it surfaces above negation.

Dutch *ieder* (‚every‘) may not reconstruct below negation.

# XIII. PPI-exhaustification

- But why would weak PPIs be fine with taking scope under higher, but not under lower anti-licensers?

- And why can strong PPIs not take scope under higher and under lower anti-licensers?

# XIII. PPI-exhaustification

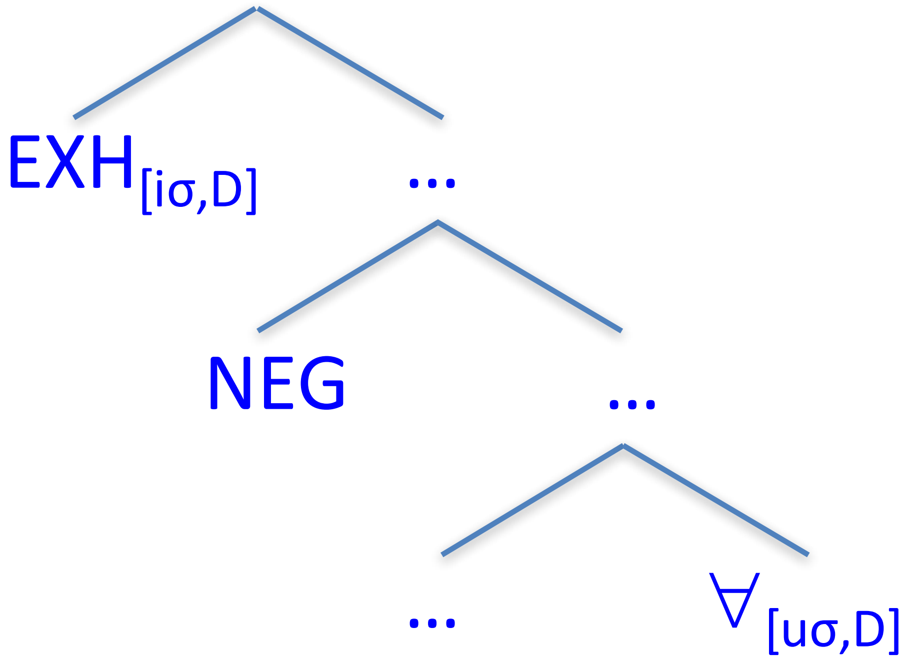
- Solution: EXH(+)>NEG>∀ yields a contradiction. But NEG>EXH(+)>∀ does not!

- So, it al depends on where EXH(+) is present in the structure.

- Covert $EXH+_{[i\sigma,D]}$ is always higher than the PPI at surface structure, since it must appear in a position c-commanding its syntactic feature (cf. Zeijlstra 2004, 2012).

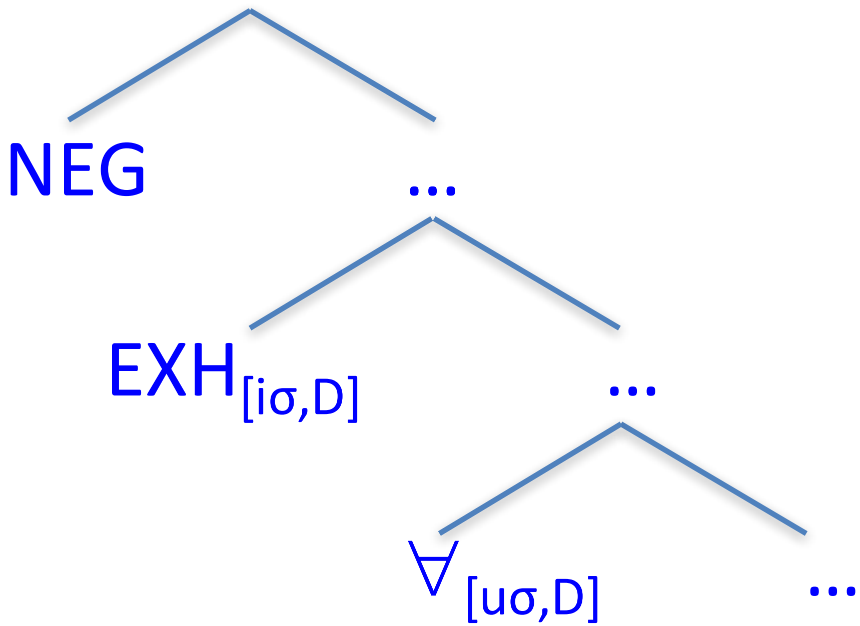
# XIII. PPI-exhaustification

- When the PPI appears below negation, one parse gives rise to a contradiction:



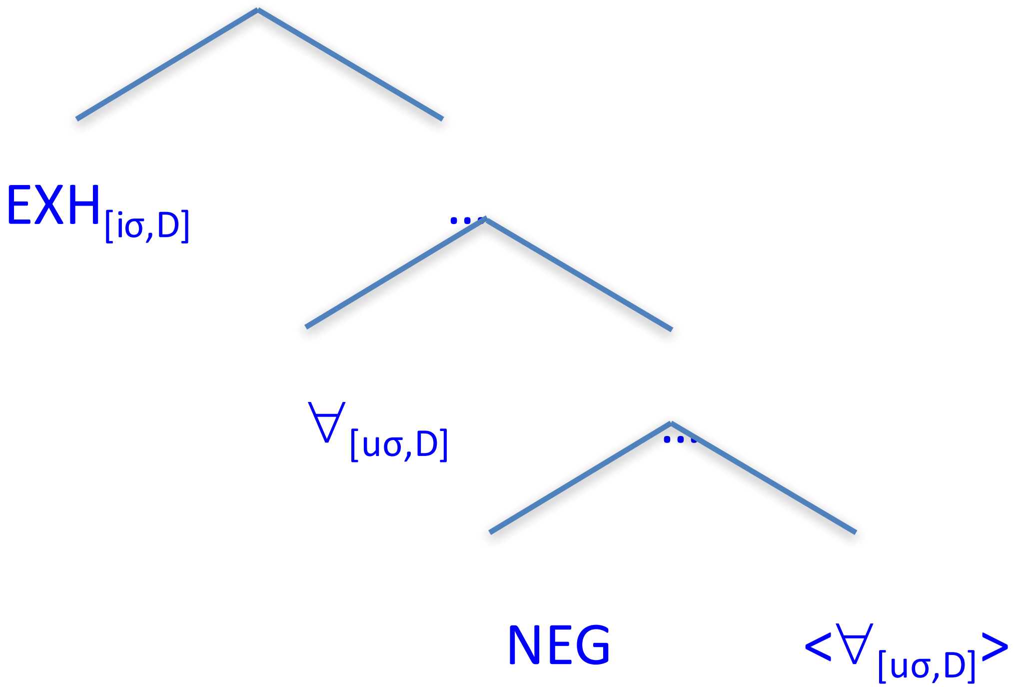
# XIII. PPI-exhaustification

- But the sentence also allows a parse that does not give rise to a contradiction. Hence the PPI may appear below negation:



# XIII. PPI-exhaustification

- When the PPI precedes negation, no contradiction arises either; but the contradiction would arise if the PPI reconstructed below negation:



# XIII. PPI-exhaustification

- Syntactically exhaustified Universal Quantifier PPIs can scope under negation, as long as the exhaustifier is able to intervene between the negation (or another anti-licenser) and the PPI.

- This way the linear-sensitivity effects attested for weak Universal Quantifier PPIs follow.

# XIII. PPI-exhaustification

- Weak existential NPIs are NPIs for the same reason as why strong universal PPIs are PPIs: only the assertion is evaluated for being DE under exhaustification.

- Strong existential NPIs are NPIs for the same reason as why weak universal PPIs are PPIs: all meaning levels are evaluated for being DE under exhaustification.

# XIII. PPI-exhaustification

- Recall our hypothesis:

- EXH+: an exhaustifier that can check off (by carrying [iσ,D]) a feature on an element carrying [uσ,D].

- EXH: an exhaustifier that can only be introduced as a last resort if a grammatical sentence (i.e. a sentence with no unchecked features) still contains unaffected domain alternatives.

# XIII. PPI-exhaustification

- Syntactic exhaustification:
- is triggered by Agree;
- is subject to syntactic locality constraints;
- may apply at any position in the clause, provided its complement is of the right semantic type and it c-commands the PI;
- Involves EXH+.

- Pragmatic exhaustification:
- takes place as a last resort operation;
- is not subject to syntactic locality constraints;
- may apply at the CP level only (given that it is a last resort operation applying at clausal level);
- Involves EXH.

# XIII. PPI-exhaustification

- Only syntactic exhaustification can give rise to linear-sensitive effects on PPIs.

- Syntactic exhaustification involves EXH+.

- Exhaustification by EXH+ only applies to strong NPIs and weak PPIs.

# XIII. PPI-exhaustification

- Weak NPIs and Strong PPIs are exhaustified by EXH.

- Exhaustification by EXH involves pragmatic exhaustification.

- Pragmatic exhaustification only applies at CP-level and can thus not give rise to linear-sensitivity effects on PPIs.

# XIV. EXH vs EXH+

- The evidence from PPIs confirms our hypothesis and points in the direction of EXH+ being triggered syntactically and EXH triggered pragmatically.

- But why would syntactic exhaustification involve pragmatic evaluation and pragmatic exhaustification involve purely semantic evaluation?

# XIV. EXH vs EXH+

- Note that *a prima facie* the reverse seems more intuitive (although other syntactic operators that take enriched meaning into consideration do exist).

- However, there may be reasons why the picture is as it is.

- First, one should distinguish between *exhaustification as an operator* vs *exhaustification as an operation*.

# XIV. EXH vs EXH+

- EXH seems a typical Neo-Gricean operation. It derives a meaning enrichment taking the assertion as its input. In that sense the fact that EXH involves only evaluation of the assertion can be motivated.

- But why does syntactic exhaustification then involve pragmatic evaluation (evaluation of enriched meanings)?

# XIV. EXH vs EXH+

- EXH+ is a lexical item. It is therefore a real operator.

- Speculation #1: EXH+ involves enriched meaning evaluation as it can only be functionally motivated if it is different from the pragmatic operation EXH.

# XIV. EXH vs EXH+

- Speculation #2: EXH+ allows pragmatic evaluations to be applied inside the clause (and not after the clause, or even the sentence, is fully generated).

- It is purely grammaticalized pragmatic evaluation in that sense.

# XV. Conclusions

- Strong NPIs and weak PPIs trigger syntactic exhaustification (along the lines of Chierchia 2006, 2013). Weak NPIs and strong PPIs trigger pragmatic exhaustification (along the lines of Krifka 1995).

- Syntactic exhaustification involves EXH+ (exhaustification of enriched meanings). Pragmatic exhaustification involves EXH (exhaustification of assertions).

# XV. Conclusions

- These hypotheses make correct predictions in the domain of universal quantifier PPIs: weak universal quantifier PPIs are predicted to exhibit linear-sensitivity effects; strong universal quantifier PPIs are not.

- The difference between pragmatically triggered EXH and syntactically triggered EXH+ may arguably be understood in terms of global pragmatic operations vs local pragmatic operators.

# Thanks!