

Negative Concord in Romanian as Polyadic Quantification

Gianina Iordăchioaia

Universität Stuttgart

Frank Richter

Universität Tübingen

Proceedings of the 16th International Conference on
Head-Driven Phrase Structure Grammar

Georg-August-Universität Göttingen, Germany


Stefan Müller (Editor)

2009

CSLI Publications

pages 150–170

<http://csli-publications.stanford.edu/HPSG/2009>

Iordăchioaia, Gianina, & Richter, Frank. 2009. Negative Concord in Romanian as Polyadic Quantification. In Müller, Stefan (Ed.), *Proceedings of the 16th International Conference on Head-Driven Phrase Structure Grammar, Georg-August-Universität Göttingen, Germany*, 150–170. Stanford, CA: CSLI Publications. 

Abstract

In this paper we develop an HPSG syntax-semantics of negative concord in Romanian. We show that n-words in Romanian can best be treated as negative quantifiers which may combine by resumption to form polyadic negative quantifiers. Optionality of resumption explains the existence of simple sentential negation readings alongside double negation readings. We solve the well-known problem of defining general semantic composition rules for translations of natural language expressions in a logical language with polyadic quantifiers by integrating our higher-order logic in Lexical Resource Semantics, whose constraint-based composition mechanisms directly support a systematic syntax-semantics for negative concord with polyadic quantification.

1 Introduction

We present an analysis of the syntax and semantics of the core of Romanian Negative Concord (NC) constructions as polyadic quantification in Lexical Resource Semantics (LRS, Richter and Sailer (2004)). Following a proposal by de Swart and Sag (2002) for French, we express the truth conditions associated with Romanian NC constructions by means of negative polyadic quantifiers. Going beyond de Swart and Sag’s largely informal treatment of the logical representations for polyadic quantification in HPSG, we extend the logical representation language and modify the interface principles of LRS to accommodate polyadic quantifiers. This way we arrive at a theory of Romanian NC using resumptive polyadic quantifiers. Resumptive polyadic quantifiers are a notorious problem for frameworks which use the lambda calculus in combination with a functional theory of types to define a compositional semantics for natural languages. Our proposal of implementing them with LRS overcomes these fundamental logical limitations, and LRS is powerful enough to specify by standard HPSG devices a precise systematic relationship between a surface-oriented syntax and semantic representations with polyadic quantifiers.

Sentential negation in Romanian is usually expressed by the verbal prefix *nu* (Barbu (2004)). In the absence of other negative elements, *nu* contributes semantic negation (1a). If in addition an n-word such as *niciun* is present (1b), only a negative concord (NC) reading is available, a double negation (DN) interpretation is not. The negation marker (NM) *nu* is obligatory with n-words. In constructions with two n-words, both a NC reading and a DN reading are available (1c).¹

[†]We would like to thank Janina Radó for proofreading and many suggestions. We also thank Danièle Godard, Doug Arnold and the audience of HPSG09 for stimulating comments and discussion.

¹The DN reading in (1c) is dependent on a context in which one speaker formulates a negative proposition using the n-constituent *nicio carte* and another speaker denies that proposition by means of the n-constituent *niciun student*. See Iordăchioaia (2009, §3.4.2) for details.

- (1) a. Un student **nu** a venit.
a student NM has come
‘Some student didn’t come.’
- b. **Niciun** student *(**nu**) a venit.
no student NM has come
i. ‘No student came.’ (NC)
ii. # ‘No student didn’t come.’ (DN)
- c. **Niciun** student **nu** a citit **nicio** carte.
no student NM has read no book
i. ‘No student read any book.’ (NC)
ii. ‘No student read no book.’ (DN)

NC poses an immediate problem for composing the meaning of sentences from the meaning of their parts: Several apparently negative constituents are ultimately interpreted as single sentential negation. “NPI approaches” to NC solve this puzzle by postulating that n-words like the ones in (1b) and (1c) are in fact negative polarity items (NPIs) without inherent semantic negation (Ladusaw (1992)). Such theories, however, cannot account for the DN reading in (1c). (1c) together with (1b) suggests that (a) n-words are exponents of semantic negation, and (b) the negative marker *nu* does not contribute negation in the presence of n-words. As one of its main features, our syntax-semantics interface for Romanian NC acknowledges the lexically negative semantics of n-words and of the NM, and it captures under what circumstances the inherent negativity of the NM can be observed.

The remainder of the paper is structured as follows: First we discuss the data that lead us to conclude that Romanian n-words are indeed negative quantifiers (Section 2). Then we move on to the tools that we need to formulate our theory and extend the logical object language and the principles of LRS in such a way as to have resumptive polyadic quantifiers at our disposal (Section 3). The core of our theory of Romanian NC is presented in Section 4, where we formulate a language-specific principle that captures the properties of simple Romanian NC constructions. In Section 5 we show that our analysis can be extended in a straightforward way to more complex cases which involve scope properties of negative quantifiers in embedded subjunctive clauses. In the final section we briefly summarize the results and speculate about possible future developments.

2 Data

In this section we discuss evidence for the negative semantics of Romanian n-words and for their quantificational behavior. We focus on the properties of n-words in Romanian and on counterevidence for a treatment of Romanian n-words as NPIs. Alternative approaches to NC will not be considered here; a detailed discussion can be found in Iordăchioaia (2009).

NPI approaches to NC rest on two claims: (a) n-words lack negation, and (b) they are semantically licensed by an anti-additive operator (see below for an algebraic characterization of anti-additivity). Ladusaw (1992) argues that the semantic licenser of NPIs may be covert. This proposal has been widely exploited in the minimalist tradition (see, for instance, Zeijlstra (2004)), but is not available in a surface-oriented syntactic framework such as HPSG. Without the option of an empty syntactic operator, the only plausible licenser of n-words in a NC construction like (1b) is the NM. In Romanian the NM is usually obligatory with n-words, which has been interpreted as a consequence of its function as a semantic licenser. Analyses that adopt this view were formulated for Polish NC in Przepiórkowski and Kupść (1999) and Richter and Sailer (1999), and for Romanian in Ionescu (1999). We do not subscribe to this idea and will show instead that although the Romanian NM acts as a licenser for NPIs, it does not behave like a semantic licenser for n-words, and n-words do not need a semantic licenser, as they carry negation themselves.

According to Ladusaw, the semantic licenser of n-words must be at least anti-additive. A negative function f is anti-additive iff for each pair of sets X and Y , $f(X \cup Y) = f(X) \cap f(Y)$. In the absence of n-constituents, the NM *nu* receives an anti-additive interpretation (2):

- (2) a. Studenții **nu** au citit romane *sau* poezii.
 students-the NM have read novels or poems
 ‘The students haven’t read novels or poems.’
 b. = Studenții **nu** au citit romane *și* studenții **nu** au
 students-the NM have read novels and students-the NM have
 citit poezii.
 read poems
 = ‘The students haven’t read novels and the students haven’t read poems.’

If the disjunction that *nu* takes as argument contains n-words, anti-additivity disappears, and the two n-words are interpreted independently under the scope of negation (3):

- (3) a. Studenții **nu** au citit **niciun** roman *sau* **nicio** poezie.
 students-the NM have read no novel or no poem
 ‘The students read no novel or no poem.’
 b. \neq Studenții **nu** au citit **niciun** roman *și* studenții
 students-the NM have read no novel and students-the
nu au citit **nicio** poezie.
 NM have read no poem
 \neq ‘The students read no novel and the students read no poem.’

- c. = Studenții **nu** au citit **niciun** roman *sau* studenții
 students-the NM have read no novel or students-the
nu au citit **nicio** poezie.
 NM have read no poem
 = ‘The students read no novel or the students read no poem.’

If the n-words in (3) are replaced with NPIs, the anti-additivity test succeeds. The contrast between (3) and (4) indicates that *nu* acts as licenser for NPIs but not for n-words.

- (4) a. Studenții **nu** au citit **vreun** roman *sau* **vreo** poezie.
 students-the NM have read any novel or anyo poem
 ‘The students didn’t read any novel or any poem.’
 b. = Studenții **nu** au citit **vreun** roman *și* studenții **nu**
 students-the NM have read any novel and students-the NM
 au citit **vreo** poezie.
 have read any poem
 = ‘The students didn’t read any novel and the students didn’t read any poem.’

Evidence for the inherent negativity of n-words comes from fragmentary answers (5a) and past participial constructions (5b), where n-words do not require the presence of the NM and contribute negation alone:

- (5) a. A: Who was at the door?
 B: **Nimeni**.
 nobody
 b. articol **de nimeni** citat
 article by nobody cited
 ‘article which hasn’t been cited by anybody’

In these contexts n-words exhibit anti-additivity (6), and they can also license NPIs. The NPI *vreo* can be licensed by the anti-additive n-word *nimeni* but not by the universal quantifier *toată* (7).

- (6) a. A: Who was at the door?
 B: **Nimeni** cunoscut *sau* important. = **Nimeni** cunoscut *și*
 nobody known or important nobody known and
nimeni important.
 nobody important
 b. articol [de **nimeni** citat *sau* lădat] = articol [de **nimeni** citat
 article by nobody cited or praised article by nobody cited
și de **nimeni** lădat]
 and by nobody praised

‘article which hasn’t been cited or praised by anybody’

- (7) articol [de **nimeni**/*de toată lumea citat la *vreo* conferință]
article by nobody/by all people cited at any conference
‘article which hasn’t been cited by anybody at any conference’

The data in (6) and (7) clearly show that n-words carry negative semantics, which is hard to reconcile with the idea that they need a semantic licenser. Besides their negative content, n-words display scope properties that are similar to those of bona fide quantifiers and contrast with those of NPIs. We observe that n-words can build NC with a NM across a subjunctive clause boundary (8a), but not across a ‘that’ complementizer (8b). This behavior is paralleled by universal quantifiers, which can take wide scope over an operator in the matrix clause from an embedded subjunctive clause (9a), but not from an embedded ‘that’-clause (9b).

- (8) a. Ion **nu** a încercat să citească **nicio** carte.
John NM has tried SJ read no book
‘John didn’t try to read any book.’
b. Ion **nu** a zis că a citit *vreo*/***nicio** carte.
John NM has said that has read any/no book
- (9) a. *Un* student a încercat să citească **fiecare** carte.
a student has tried SJ read every book
‘Some student tried to read every book.’
i. $\exists > \forall$; ii. $\forall > \exists$
b. *Un* student a zis că a citit **fiecare** carte.
a student has said that has read every book
‘Some student said that s/he read every book.’
i. $\exists > \forall$; ii. $\# \forall > \exists$

In addition, adjunct clauses and relative clauses block NC formation (10) and wide scope of embedded universal quantifiers (11), but not NPI licensing (10):

- (10) a. **Nu** am dezvăluit secrete [care să-l fi expus pe
NM have revealed secrets that SJ-CL be exposed PE
***niciun**/*vreun* coleg].
no/any colleague
‘I didn’t reveal secrets that exposed any colleague.’
b. **Nu** am spus asta [pentru că mi-o ceruse ***niciun**/*vreun*
NM have said this because CL-CL asked no/any
prieten].
friend
‘I didn’t say that because any friend had asked me to.’

- (11) a. *Un student a dezvăluit secrete [care l-au expus pe
a student has revealed secrets that CL-have exposed PE
fiecare coleg].*
every colleague
'Some student revealed secrets that exposed every colleague.'
i. $\exists > \forall$; ii. $\# \forall > \exists$
- b. *Un student a spus asta [pentru că i-o ceruse **fiecare**
a student has said this because CL-CL asked every
prieten].*
friend
'Some student said that because every friend had asked him to.'
i. $\exists > \forall$; ii. $\# \forall > \exists$

The negative semantics and the quantificational properties of n-words explain the possibility of a DN reading with two n-words in (1c). The DN reading is the interpretation we expect with two negative quantifiers. In this respect there is no difference between the semantic status of n-words in Romanian and in DN languages like standard English or German, where DN is the only interpretation for two co-occurring n-constituents. What remains to be explained is the availability of the NC reading in (1c).

Following de Swart and Sag (2002), we analyze determiner n-words and negative NP constituents as quantifiers of Lindström type $\langle 1, 1 \rangle$ and $\langle 1 \rangle$, respectively (see Lindström (1966)). They may combine by resumption to form a polyadic quantifier of type $\langle 1^n, n \rangle$ or $\langle n \rangle$ (van Benthem (1989), Keenan and Westerstahl (1997), Peters and Westerstahl (2006)) and thus give rise to an NC interpretation. The negative marker *nu* is analyzed as a negative quantifier of type $\langle 0 \rangle$ that is absorbed under resumption with other negative polyadic quantifiers. The relevant technical details will be sketched in our LRS implementation of polyadic quantification and resumption below.

3 LRS with Polyadic Quantifiers

For our analysis we need a higher-order logical language with negative polyadic quantifiers. Here we briefly outline its crucial properties and indicate how to integrate it with LRS.

We assume a simple type theory with types e and t . Functional types are formed in the usual way. The syntax of the logical language provides function application, lambda abstraction, equality and negative polyadic quantifiers. By standard results this is enough to express the usual logical connectives and monadic quantifiers. In reference to the simple type theory, we call our family of languages Ty1. *Var* and *Const* are a countably infinite supply of variables and constants of each type:

Definition 1 Ty1 Terms: *Ty1* is the smallest set such that:

$Var \subset Ty1$, $Const \subset Ty1$,

for each $\tau, \tau' \in Type$, for each $\alpha_{\tau\tau'}, \beta_{\tau} \in Ty1$:

$$(\alpha_{\tau\tau'}\beta_{\tau})_{\tau'} \in Ty1,$$

for each $\tau, \tau' \in Type$, for each $i \in \mathbb{N}^+$, for each $v_{i,\tau} \in Var$, for each $\alpha_{\tau'} \in Ty1$:

$$(\lambda v_{i,\tau}.\alpha_{\tau'})_{(\tau\tau')} \in Ty1,$$

for each $\tau \in Type$, and for each $\alpha_{\tau}, \beta_{\tau} \in Ty1$:

$$(\alpha_{\tau} = \beta_{\tau})_t \in Ty1,$$

for each $\tau \in Type$, for each $n \in \mathbb{N}^0$, for each $i_1, i_2, \dots, i_n \in \mathbb{N}^+$, for each $v_{i_1,\tau}, v_{i_2,\tau}, \dots, v_{i_n,\tau} \in Var$, for each $\alpha_{t1}, \alpha_{t2}, \dots, \alpha_{tn}, \beta_t \in Ty1$:

$$(NO(v_{i_1,\tau}, \dots, v_{i_n,\tau})(\alpha_{t1}, \dots, \alpha_{tn})(\beta_t))_t \in Ty1.$$

The standard constructs receive their usual interpretation. Here we only state the interpretation of negative polyadic quantifiers:

Definition 2 The Semantics of Ty1 Terms

(clause for negative polyadic quantifiers only)

For each model M and for each variable assignment $a \in Ass$, for each $\tau \in Type$, for each $n \in \mathbb{N}^0$, for each $i_1, i_2, \dots, i_n \in \mathbb{N}^+$, for each $v_{i_1,\tau}, v_{i_2,\tau}, \dots, v_{i_n,\tau} \in Var$, for each $\alpha_{t1}, \alpha_{t2}, \dots, \alpha_{tn}, \beta_t \in Ty1$:

$$\llbracket NO(v_{i_1,\tau}, \dots, v_{i_n,\tau})(\alpha_{t1}, \dots, \alpha_{tn})(\beta_t) \rrbracket^{M,a} = 1 \text{ iff}$$

for every $d_{i_1}, d_{i_2}, \dots, d_{i_n} \in D_{E,\tau}$,

$$\llbracket \alpha_{t1} \rrbracket^{M,a[v_{i_1,\tau}/d_{i_1}]} = 0 \text{ or } \llbracket \alpha_{t2} \rrbracket^{M,a[v_{i_2,\tau}/d_{i_2}]} = 0 \text{ or } \dots$$

$$\text{or } \llbracket \alpha_{tn} \rrbracket^{M,a[v_{i_n,\tau}/d_{i_n}]} = 0 \text{ or } \llbracket \beta_t \rrbracket^{M,a[(v_{i_1}, \dots, v_{i_n})/(d_{i_1}, \dots, d_{i_n})]} = 0.$$

(12) shows the truth conditions that we obtain for the translation of the Romanian counterparts of *John didn't come* (12a) and *No teacher didn't give no book to no student*, where all NPs are n-constituents and form a ternary negative quantifier by resumption (12b):

- (12) a. For $n = 0$, $\llbracket NO()(\text{come}'(j)) \rrbracket^{M,a} = 1$ iff $\llbracket \text{come}'(j) \rrbracket^{M,a} = 0$
 b. For $n = 3$, $v_{i_1} = x, v_{i_2} = y, v_{i_3} = z$, $\alpha_{t1} = \text{teacher}'(x)$,
 $\alpha_{t2} = \text{book}'(y)$, $\alpha_{t3} = \text{student}'(z)$ and $\beta_t = \text{give}'(x, y, z)$,
 $\llbracket NO(x, y, z)(\text{teacher}'(x), \text{book}'(y), \text{student}'(z))$
 $(\text{give}'(x, y, z)) \rrbracket^{M,a} = 1$ iff for every $d_1, d_2, d_3 \in D_{E,e}$,

$$\begin{aligned}
& \llbracket teacher'(x) \rrbracket^{M,a[x/d_1]} = 0 \text{ or } \llbracket book'(y) \rrbracket^{M,a[y/d_2]} = 0 \text{ or} \\
& \llbracket student'(z) \rrbracket^{M,a[z/d_3]} = 0 \text{ or} \\
& \llbracket give'(x, y, z) \rrbracket^{M,a[(x,y,z)/(d_1,d_2,d_3)]} = 0
\end{aligned}$$

Minor adjustments suffice to integrate these logical representations in LRS. In the signature, the appropriateness of *gen-quantifier* of Richter and Kallmeyer (2009) is generalized to lists of variables (instead of single variables), and the restrictor of quantifiers now contains a list of expressions:

```

me  TYPE  type
    gen-quantifier  VAR  list
                        RESTR  list
                        SCOPE  me

```

A new statement in the theory of well-formed logical expressions (13) restricts polyadic generalized quantifiers to the form given in DEFINITION 1. The four relations mentioned in (13) are defined in such a way that they guarantee that $\boxed{1}$ is a list of variables, all variables have the same type $\boxed{3}$, the expressions in the list of restrictors $\boxed{2}$ are of type t , and there are exactly as many restrictor expressions as variables:

$$\begin{aligned}
(13) \quad gen\text{-}quantifier \rightarrow & \left[\begin{array}{l} \text{TYPE } \textit{truth} \\ \text{VAR } \boxed{1} \\ \text{RESTR } \boxed{2} \\ \text{SCOPE} | \text{TYPE } \textit{truth} \end{array} \right] \\
& \wedge \text{variable-list}(\boxed{1}) \wedge \text{same-type-list}(\boxed{3}, \boxed{1}) \\
& \wedge \text{truth-list}(\boxed{2}) \wedge \text{same-length}(\boxed{1}, \boxed{2})
\end{aligned}$$

We follow the usual notational conventions in LRS and often write descriptions of expressions of the semantic representation language as (partial) logical expressions. For describing polyadic quantifiers we use the notation $Q(\vec{v}, \vec{\phi}, \psi)$. Here \vec{v} and $\vec{\phi}$ are shorthand for a (possibly empty) list of variables and a (possibly empty) list of expressions; ψ is a single expression. In the analysis of Romanian below we will assume that there is an appropriate subsort of *gen-quantifier* in our grammar which is interpreted as negative polyadic quantifier. In our notation this family of quantifiers will be denoted by $no(\vec{v}, \vec{\phi}, \psi)$.

The clause of the SEMANTICS PRINCIPLE governing the combination of quantificational determiners with nominal heads has to be adjusted to polyadic quantifiers. The relevant clause is shown in (14). Except for the generalization from monadic quantifiers to polyadic quantifiers, it is identical to the corresponding clause in (Richter and Kallmeyer, 2009, p. 65).

- (14) THE SEMANTICS PRINCIPLE, Clause 1
 If the non-head is a quantifier, then its INCONT value is of the form $Q(\vec{v}, \vec{\phi}, \psi)$, the INCONT value of the head is a component of a member²

²The symbol “ \triangleleft_{\in} ” is the infix notation of the new relation *subterm-of-member*, a generalized subterm relation.

of the list $\vec{\phi}$, and the INCONT value of the non-head daughter is identical to the EXCONT value of the head daughter:

$$\left[\begin{array}{c} \text{DTRS} | \text{SPR-DTR} | \text{SS} | \text{LOC} \left[\begin{array}{c} \text{CAT} | \text{HEAD} \quad \text{det} \\ \text{CONT} | \text{MAIN} \quad \text{gen-quantifier} \end{array} \right] \end{array} \right] \rightarrow$$

$$\left(\left[\begin{array}{c} \text{DTRS} \left[\begin{array}{c} \text{H-DTR} | \text{LF} \left[\begin{array}{c} \text{EXCONT} \boxed{1} \\ \text{INCONT} \boxed{2} \end{array} \right] \\ \text{SPR-DTR} | \text{LF} \left[\begin{array}{c} \text{INCONT} \boxed{1} \left[\begin{array}{c} \text{gen-quantifier} \\ \text{RESTR} \boxed{3} \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right] \wedge \boxed{2} \triangleleft \boxed{3} \right)$$

Resumption will be implemented in LRS as identity of quantifiers contributed by lexical elements. For that reason no special technical apparatus for the resumption operation has to be introduced in preparation of our analysis of negative concord in Romanian in the next section.

With the integration of polyadic quantifiers and the modified clause of the SEMANTICS PRINCIPLE we have completed the adjustments in LRS needed to formulate our theory of NC. Before we turn to the analysis in the next section, we briefly review three standard LRS principles that will play a role in our examples. These are the LRS PROJECTION PRINCIPLE, the INCONT PRINCIPLE and the EXCONT PRINCIPLE. The LRS PROJECTION PRINCIPLE governs the relationship of the attribute values of EXCONT, INCONT and PARTS at phrases relative to their syntactic daughters. It is responsible for EXCONT and INCONT identity along syntactic head projections, and for the inheritance of the elements of PARTS lists by phrases from their daughters:

- (15) LRS PROJECTION PRINCIPLE (Richter and Kallmeyer, 2009, pp. 47–48)
- In each *phrase*,
1. the EXCONT values of the head and the mother are identical,
 2. the INCONT values of the head and the mother are identical,
 3. the PARTS value contains all and only the elements of the PARTS values of the daughters.

The INCONT PRINCIPLE and the EXCONT PRINCIPLE constrain the admissible values of the INCONT and the EXCONT attribute in syntactic structures. The INCONT PRINCIPLE is the simpler one of them. It guarantees two things: First, the internal content of a sign (the part of its semantics that is outscoped by any operator the sign combines with along its syntactic projection) is always semantically contributed by the sign, i.e. it is a member of its PARTS list. And second, the internal content is in the external content of a sign. In a first approximation (which is precise enough for our purposes) this means that the internal content contributes its semantics within the maximal syntactic projection of a sign.

- (16) The INCONT PRINCIPLE (Richter and Kallmeyer, 2009, p. 47)
- In each *lrs*, the INCONT value is an element of the PARTS list and a component of the EXCONT value.

The EXCONT PRINCIPLE is slightly more complex. Its first clause requires that the external content of a non-head daughter be semantically contributed from within the non-head-daughter. The second clause is a closure principle and says that the semantic representation of an utterance comprises all and only those pieces of semantic representations that are contributed by the lexical items in the utterance.

- (17) The EXCONT PRINCIPLE (Richter and Kallmeyer, 2009, p. 47)
- Clause 1:
In every phrase, the EXCONT value of the non-head daughter is an element of the non-head daughter's PARTS list.
- Clause 2:
In every utterance, every subexpression of the EXCONT value of the utterance is an element of its PARTS list, and every element of the utterance's PARTS list is a subexpression of the EXCONT value.

The effects of these principles will be relevant for the examples in the next two sections.

4 The Analysis of Romanian NC

We will proceed in two steps. In Section 4.1 we lay out the analysis of sentential negation with the verbal prefix *nu* using a lexical rule. In Section 4.2 we turn to NC in simple sentences.

4.1 Sentential Negation

The analysis of simple negated sentences without n-constituents like (1a) follows immediately from the lexical analysis of verbs with the NM prefix *nu*. The affixal nature of *nu* is extensively argued for in Barbu (2004). Following assumptions similar to ours in Ionescu (1999) and the parallel analysis of the Polish negative marker in Przepiórkowski and Kupść (1997), we formulate the lexical rule in (18) that relates each verb form of the appropriate kind to a corresponding negated form.

(18) THE NM LEXICAL RULE

$$\left[\begin{array}{l} \text{word} \\ \text{PHON } [4] \\ \text{SS|LOC|CAT} \left[\begin{array}{l} \text{HEAD} \left[\begin{array}{l} \text{verb} \\ \text{VFORM } fin \vee inf \\ \text{NEG } - \end{array} \right] \\ \text{EXCONT } [0] \\ \text{LF } \left[\begin{array}{l} \text{INCONT } [1] \\ \text{PARTS } [2] \end{array} \right] \end{array} \right] \end{array} \right] \mapsto \left(\begin{array}{l} \text{PHON } Neg([4]) \\ \text{SS | LOC | CAT | HEAD } [NEG \ +] \\ \text{LF | PARTS } [2] \oplus \langle [3] no(\vec{u}, \vec{\gamma}, \delta) \rangle \\ \wedge [1] \triangleleft \delta \wedge [3] \triangleleft [0] \end{array} \right)$$

The NM attaches to finite and infinitival verb forms as indicated by the VFORM value in (18). The boolean NEG feature value ensures that the NM is attached to a verb only once. All verb forms in the lexicon are specified as [NEG −] and may have a [NEG +] counterpart only if they undergo the lexical rule. The function *Neg* in the PHON value description of the output is responsible for the correct phonological forms with the verbal prefix. It permits reduction of *nu* to *n-* depending on the first phoneme in the input’s verb form.

The semantic counterpart to the prefix *nu* in the phonological form is a negative quantifier on the verb’s PARTS list, marked by the tag 3 in the lexical rule. The interpretation of the verb form as negated is a consequence of the requirement that the internal content of the verb 1 be a subterm of the nuclear scope δ of this quantifier ($1 \triangleleft \delta$ in the output description of the lexical rule). The negative quantifier 3 is also a subterm of the external content 0 of the verb ($3 \triangleleft 0$). This condition will become important in the analysis of embedded clauses in Section 5 and will be responsible for the inability of the negation on an embedded verb form to outscope a matrix verb. As we will see later, negative quantifiers contributed by n-words in argument position will, under certain conditions, have the option of taking wide scope from embedded clauses.

The negative verb form *nu a venit* in our sentence (1a) is licensed by the NM LEXICAL RULE and shown below:

(19) *nu a venit* (‘NM has come’, licensed by the NM LEXICAL RULE)

$$\left[\begin{array}{l} \text{word} \\ \text{PHON} \langle \text{nu}, \text{a}, \text{venit} \rangle \\ \text{SS} \mid \text{LOC} \left[\begin{array}{l} \text{CAT} \left[\begin{array}{l} \text{HEAD} \mid \text{NEG} + \\ \text{VAL} \mid \text{SUBJ} \langle \text{NP}_{1a} \rangle \end{array} \right] \\ \text{CONT} \left[\begin{array}{l} \text{INDEX} \mid \text{VAR} \text{ no-var} \\ \text{MAIN} \quad 3a \text{ come}' \end{array} \right] \end{array} \right] \\ \text{LF} \left[\begin{array}{l} \text{EXCONT} \quad 0 \\ \text{INCONT} \quad 3 \text{ come}' (1a) \\ \text{PARTS} \quad \langle 3, 3a, 7 \text{ no}(\vec{u}, \vec{\gamma}, \delta) \rangle \end{array} \right] \end{array} \right] \wedge 3 \triangleleft 0 \wedge 3 \triangleleft \delta \wedge 7 \triangleleft 0$$

With standard LRS mechanisms in combination with a language-specific constraint that excludes the existential quantifier originating from *un student* from occurring in the immediate scope of negation, we obtain $\text{some}(x, \text{student}'(x), \text{no}((\cdot), (\cdot), \text{come}'(x)))$ as the truth condition for (1a). The variable and restrictor lists of the negative quantifier are empty (Lindström type $\langle 0 \rangle$) because the negative verb does not introduce a variable, and the sentence does not provide a restrictor.

4.2 NC Constructions

Determiner n-words contribute negative quantifiers of underspecified Lindström type $\langle 1^n, n \rangle$. In their LRS representation they lexically contribute exactly one new

(20) a.
$$\left[\begin{array}{l} \text{word} \\ \text{PHON } \langle \text{niciun} \rangle \\ \text{SS } | \text{ LOC } \left[\begin{array}{l} \text{CAT } | \text{ HEAD } \left[\begin{array}{l} \text{det} \\ \text{SPEC } \text{N} \end{array} \right] \\ \text{CONT } \left[\begin{array}{l} \text{INDEX } | \text{ VAR } \end{array} \right] \\ \text{MAIN } \end{array} \right] \\ \text{LF } \left[\begin{array}{l} \text{lrs} \\ \text{EXC } \text{me} \\ \text{INC } \\ \text{PARTS } \end{array} \right] \end{array} \right] \wedge x \in \vec{v} \wedge x \triangleleft_{\in} \vec{\alpha} \wedge x \triangleleft \beta$$

b.
$$\left[\begin{array}{l} \text{word} \\ \text{PHON } \langle \text{student} \rangle \\ \text{SS } | \text{ LOC } \left[\begin{array}{l} \text{CAT } \left[\begin{array}{l} \text{HEAD } \text{noun} \\ \text{VAL } | \text{ SPR } \end{array} \right] \\ \text{CONT } \left[\begin{array}{l} \text{INDEX } | \text{ VAR } \\ \text{MAIN } \end{array} \right] \end{array} \right] \\ \text{LF } \left[\begin{array}{l} \text{lrs} \\ \text{EXC } \text{gen-quantifier} \\ \text{INC } \\ \text{PARTS } \end{array} \right] \end{array} \right]$$

S

$$\left[\begin{array}{l} \text{EXCONT} \quad \boxed{0} \\ \text{INCONT} \quad \boxed{3} \\ \text{PARTS} \quad \langle \boxed{1}, \boxed{1a}, \boxed{2}, \boxed{2a}, \boxed{3}, \boxed{3a}, \boxed{7} \rangle \end{array} \right] \wedge \boxed{3} \triangleleft \beta \wedge \boxed{1} \triangleleft \boxed{0} \wedge \boxed{7} \triangleleft \boxed{0}$$

NP

$$\left[\begin{array}{l} \text{EXCONT} \quad \boxed{1} \text{ no}(\vec{v}, \vec{\alpha}, \beta) \\ \text{INCONT} \quad \boxed{2} \text{ student}'(\boxed{1a}x) \\ \text{PARTS} \quad \langle \boxed{1}, \boxed{1a}x, \boxed{2}, \boxed{2a} \text{ student}' \rangle \end{array} \right] \wedge \boxed{2} \triangleleft_{\in} \vec{\alpha}$$

niciun student

V

$$\left[\begin{array}{l} \text{EXCONT} \quad \boxed{0} \\ \text{INCONT} \quad \boxed{3} \text{ come}'(\boxed{1a}) \\ \text{PARTS} \quad \langle \boxed{3}, \boxed{3a} \text{ come}', \boxed{7} \text{ no}(\vec{u}, \vec{\gamma}, \delta) \rangle \end{array} \right]$$

nu a venit

162

According to the LRS PROJECTION PRINCIPLE, the NP inherits the INCONT value $\boxed{2}$ of its nominal head. Due to the first clause of the SEMANTICS PRINCIPLE the internal content must be a subterm of a member of the restrictor list of the quantifier ($\boxed{2} \triangleleft_{\in} \vec{\alpha}$). The EXCONT value is identified with the INCONT value $\boxed{1}$ of the determiner due to the interaction of the first clause of the EXCONT PRINCIPLE with the other restrictions on the EXCONT of the NP. At the S node of the sentence two more restrictions become relevant. All lexically introduced pieces of semantic representation must be realized in the EXCONT of the sentence, including the EXCONT of the NP and the negative polyadic quantifier from the PARTS list of the verb ($\boxed{1} \triangleleft \boxed{0}$, $\boxed{7} \triangleleft \boxed{0}$). Moreover, the standard clause of the LRS SEMANTICS PRINCIPLE for combining NP-quantifiers in argument position with verbal projections requires that the polyadic quantifier of the NP take scope over the verb ($\boxed{3} \triangleleft \beta$).

All these restrictions together license three distinct expressions in the EXCONT of the sentence. Only one of them, shown in (21a), corresponds to the linguistic facts, the other two result from possible scope interactions of the negative quantifier of the verb and the NP-quantifier. The NC reading (21a) obtains if the two negative quantifiers get identified, meaning that $\boxed{1} = \boxed{7}$, $\vec{v} = \vec{u} = x$, $\vec{\alpha} = \vec{\gamma} = student'(x)$, and $\beta = \delta = come'(x)$.

- (21) a. $no(x, student'(x), come'(x)) \boxed{0} = \boxed{1} = \boxed{7}$
b. $no(x, student'(x), no((), (), come'(x))) \boxed{0} = \boxed{1}; \boxed{3} = \delta; \beta = \boxed{7}$
c. $no((), (), no(x, student'(x), come'(x))) \boxed{0} = \boxed{7}; \boxed{3} = \beta; \delta = \boxed{1}$

(21b) and (21c) are impossible DN readings of (1b) and have to be excluded by the theory of Romanian NC. At the same time we have to take care that an n-word in a sentence obligatorily triggers the NM on the finite verb. We achieve both goals in one step by adapting the NEG CRITERION of Richter and Sailer (2004) to Romanian and the polyadic quantifier approach.

(22) THE NEG CRITERION for Romanian

If a negative quantifier of type higher than $\langle 0 \rangle$ outscopes a finite verb within the verb's external content, then the PARTS list of the verb must contain a negative quantifier of type higher than $\langle 0 \rangle$.

$$\forall \boxed{0} \forall \boxed{1} \forall \boxed{2} \left(\begin{array}{l} \left[\begin{array}{l} \text{word} \\ \text{SS} | \text{LOC} \left[\begin{array}{l} \text{CAT} | \text{HEAD} \left[\begin{array}{l} \text{verb} \\ \text{VFORM} \text{ fin} \end{array} \right] \\ \text{CONT} | \text{MAIN} \boxed{1} \end{array} \right] \\ \text{LF} | \text{EXCONT} \boxed{0} \end{array} \right] \wedge \boxed{2} no(\vec{v}, \vec{\alpha}, \beta) \triangleleft \boxed{0} \wedge \vec{v} \neq () \wedge \boxed{1} \triangleleft \beta \\ \rightarrow \exists \boxed{3} \exists \boxed{4} \left(\boxed{3} no(\vec{u}, \vec{\gamma}, \delta) \wedge \vec{u} \neq () \wedge \left[\text{LF} | \text{PARTS} \boxed{4} \right] \wedge \boxed{3} \in \boxed{4} \right) \end{array} \right)$$

Intuitively, the NEG CRITERION says that the presence of an n-word in a sentence requires the presence of a (possibly different) n-word that undergoes resump-

tion with the NM on the verb. More precisely, the NEG CRITERION is sensitive to the presence of a negative quantifier of a type higher than $\langle 0 \rangle$ in the EXCONT of a finite verb (contributed by at least one n-word). In that constellation a negative quantifier must also be on the PARTS list of the verb. Since those verbs that are licensed by lexical entries do not carry negative quantifiers in their PARTS lists, this means that only verbs licensed by the NM LEXICAL RULE are eligible. But since the quantifier contributed by a negative verb originally has an empty variable list, it would be of the excluded type $\langle 0 \rangle$ if it were not identified with a quantifier contributed by an n-word. It is due to the fact that the NEG CRITERION requires a quantifier of a type higher than $\langle 0 \rangle$ on the verb's PARTS list that identification with a quantifier from at least one n-word is necessary.

If we apply this reasoning to our example in FIGURE 1 we see that the negative quantifier contributed by the n-word and the negative quantifier on the PARTS list of the verb must be identical. We obtain an obligatory NC reading, and the other two readings in (21) are correctly ruled out.

In sentences with more than one n-word such as (1c), the negative quantifier contributed by the verb must undergo resumption with at least one of the two quantifiers contributed by the n-words for the reasons just described. If one n-word does not undergo resumption with the NM and the other n-word, we obtain the DN reading in (23a). However, there is also the possibility that all the negative quantifier contributions in the sentence are identified. The number of variables contributed by the individual n-words determines the type of the resumptive quantifier. For (1c) with two n-words, each contributing one variable, the second available alternative is resumption of all three negative quantifiers, which leads to a quantifier of type $\langle 1^2, 2 \rangle$ for the NC reading, shown in (23b).

- (23) a. $no(x, student'(x), no(y, book'(y), read'(x, y)))$ (DN)
 b. $no((x, y), (student'(x), book'(y)), read'(x, y))$ (NC)

5 N-words in Embedded Subjunctive Clauses

To complete our analysis, we investigate the function of the NM in NC constructions and show that our theory can be extended to account for locality conditions on the scope of negative quantifiers in NC constructions in complex sentences.

5.1 The NM as a Scope Marker

We argued that the NM cannot be a semantic licenser of n-words, as it does not maintain anti-additivity in the relevant contexts (3). We also saw that in NC constructions the negation contributed by the NM must always undergo resumption with at least one n-word, as decreed by the NEG CRITERION for Romanian (22). But if the NM is neither a semantic licenser, nor a real negation contributor in NC, what is its role in these constructions and why is it obligatory with n-words?

We think that an answer to these questions can be found in complex sentences like (24) where an n-word is contained in an argument phrase in an embedded subjunctive clause. In this kind of construction the negative quantifier may take wide scope over the matrix verb (24a) or narrow scope within the subjunctive clause (24b). Parallel observations hold for English n-words embedded in infinitival clauses (25). But unlike in the ambiguous English construction, in Romanian the scope of the quantifier is resolved by the (obligatory) NM: The scope of the negative quantifier is associated with the verb that carries the NM ((24a) vs. (24b)). We see that the NM functions as a *syntactic* licenser for n-words; the NM marks the sentential scope of the negative quantifier (cf. also Ionescu (1999, 2004)).

- (24) a. Ion **nu** i-a cerut Mariei [să citească **nicio** carte].
 John NM CL-has asked Mary SJ read no book
 ‘There is no book that John asked Mary to read.’
 b. Ion i-a cerut Mariei [să **nu** citească **nicio** carte].
 John CL-has asked Mary SJ NM read no book
 ‘John asked Mary not to read any book.’
- (25) I will force you to marry **no one**. (Klima (1964, p. 285))
 a. ‘I **won’t** force you to marry anyone.’
 b. ‘I would force you **not** to marry anyone.’

Assume that we augment the type theory of our semantic representation language by a type *s* for worlds and adjust the truth conditions of natural language expressions to Ty2 in the usual way. Moreover, assume for the moment that the EXCONT of matrix and embedded clause are distinct. With these modifications our theory captures (24a) and (24b).

In both sentences, independent LRS principles for quantifiers in argument position dictate that the negative quantifier associated with *nicio carte* must outscope the verb in the embedded clause. Let us look at (24a). Suppose *nicio carte* takes scope in the embedded clause. Then the NEG CRITERION is violated since the non-negated verb cannot have a negative quantifier on its PARTS list. Suppose it takes scope in the matrix clause. Then the NEG CRITERION is satisfied by resumption of the negative quantifier from *nicio carte* with the quantifier of the negated verb. We obtain the truth conditions $no(y, book'(y), ask'(john', mary', read'(mary', y)))$. The converse holds in (24b). The embedded verb has a negative marker and a negative quantifier on its PARTS, which means that *nicio carte* can take scope within the verb’s EXCONT by resumption ($ask'(john', mary', no(y, book'(y), read'(mary', y))))$. It cannot take scope in the matrix clause, because the matrix verb lacks a negative quantifier on its PARTS list.

5.2 Complex Sentences with Two NMs

The situation becomes even more complex when both the matrix and the embedded verb in a complex sentence carry a NM:

- (26) Ion **nu** i-a cerut Mariei [să **nu** citească **nicio** carte].
 John NM CL-has asked Mary SJ NM read no book
 a. ‘There is no book John asked Mary not to read.’
 b. ‘John didn’t ask Mary not to read any book.’

The sentence (26) has two readings as indicated in the two translations. The negative quantifier *nicio carte* may enter in NC with the matrix verb (26a) or with the embedded verb (26b). In either case, the other verb contributes a type $\langle 0 \rangle$ negative quantifier to the interpretation. This means that one negation outscopes the other.

In preparation of our analysis of (26), we start with the simpler case of a complex sentence without n-word but with NM at the matrix verb and the embedded verb (27). The relevant parts of its analysis tree are shown in FIGURE 2.

- (27) Ion **nu** i-a cerut Mariei [să **nu** citească *Nostalgia*].
 John NM CL-has asked Mary SJ NM read nostalgia-the
 ‘John didn’t ask Mary not to read *The Nostalgia*.’

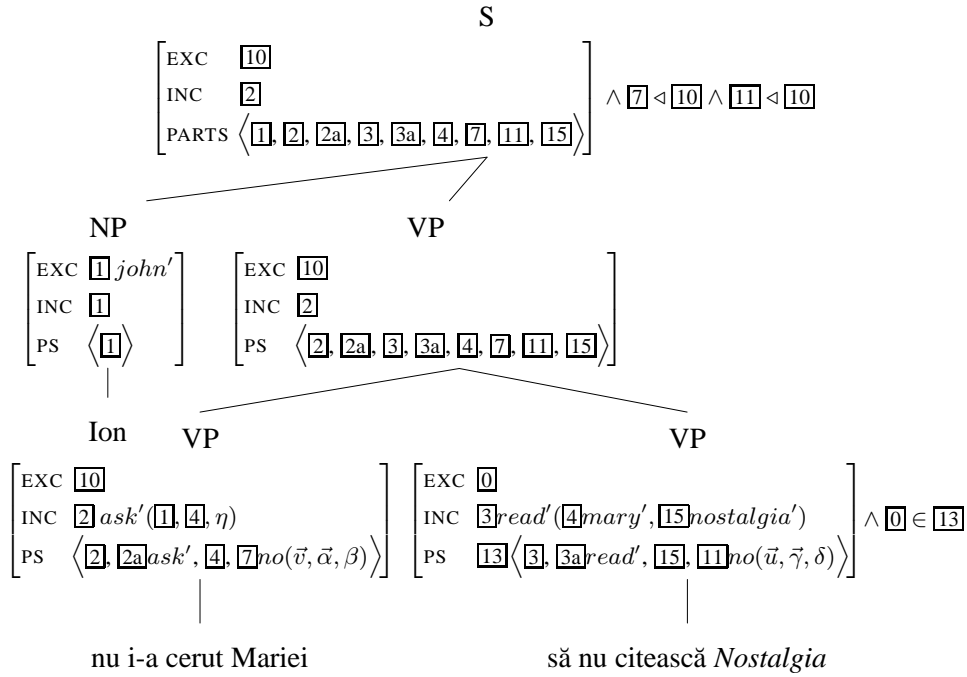


Figure 2: LRS analysis of (27) *Ion nu i-a cerut Mariei să nu citească Nostalgia*

The EXCONT of the non-head daughter VP on the right, which is the embedded subjunctive clause, must be an element of the PARTS list of that VP (EXCONT PRINCIPLE). The smallest piece of semantic representation which is eligible without violating any other LRS principles is the INCONT value $\boxed{3}$. The largest piece

of semantic representation that the EXCONT [9] of the embedded subjunctive clause can be identified with is the negative quantifier [11], which is contributed by the verb *nu citească* and is licensed by the NM LEXICAL RULE. Since the lexical rule guarantees that this negative quantifier is a subterm of the external content of the verb (see (18)), we must conclude that [9] equals [11].

It may be suprising that nothing said so far prevents the negative quantifier of the embedded verb in FIGURE 2 from taking scope in the matrix sentence. The reason is that nothing forces the quantifier [11] to take *immediate* scope over the predicate [3], the matrix predicate may intervene. As a consequence, [11] may be identified with the matrix negation or trigger DN within the matrix clause. Neither of the resulting semantic representations expresses possible truth conditions for the sentence in (27). As our analysis stands, a NM at an embedded verb could even outscope an affirmative matrix verb, giving the sentence in (28) the reading in (28b):

- (28) Ion i-a cerut Mariei [să **nu** citească *Nostalgia*].
 John CL-has asked Mary SJ NM read nostalgia-the
 a. ‘John asked Mary not to read *The Nostalgia*.’
 b. # ‘John didn’t ask Mary to read *The Nostalgia*.’

A new clause of the SEMANTICS PRINCIPLE prevents this undesired effect and ensures that the external content of the complement clause of a propositional attitude verb remains within the scope of the matrix verb:

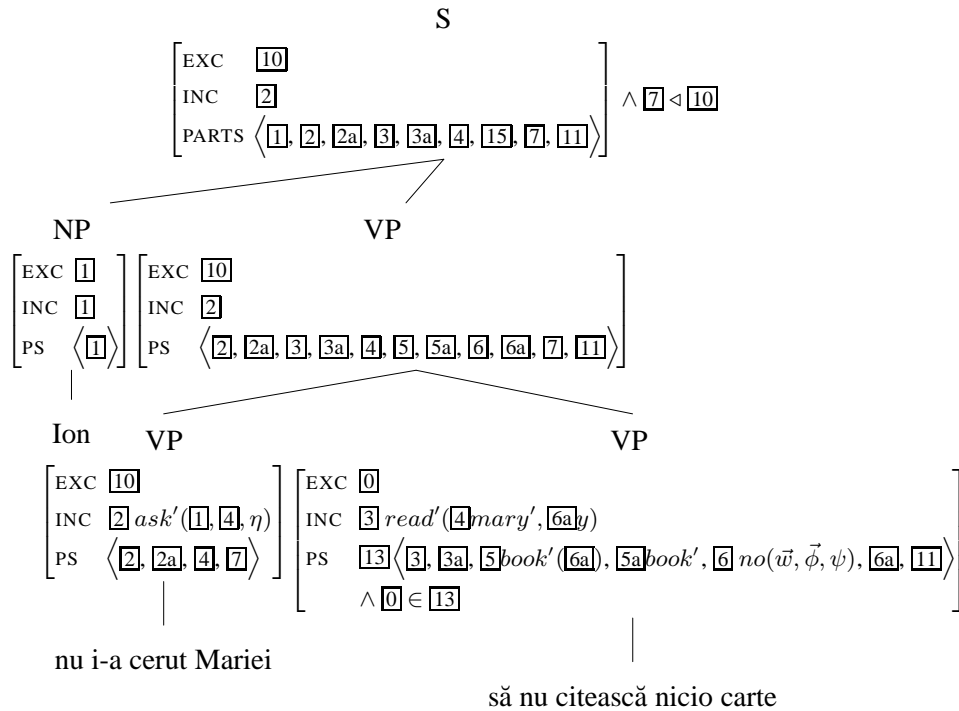
- (29) THE SEMANTICS PRINCIPLE, Clause 2
 If the head-daughter of a phrase has a MAIN value with a propositional argument η and the non-head-daughter is a propositional complement, then the EXCONT value of the complement must be a subterm of η .

In our example in FIGURE 2 the new clause of the SEMANTICS PRINCIPLE makes the EXCONT of the subjunctive clause [9] a subterm of the scope η of the verb *ask'*. The negative quantifier [11] contributed by the NM on the embedded verb is now a subterm of η and the only reading we obtain for (27) is the one in which both verbs are negated (30), as desired.

- (30) $no((), (), ask'(john', mary', no((), (), read'(mary', nostalgia'))))$

Everything is now in place for the analysis of the two readings of the ambiguous sentence (26). A description of the tree structure is given in FIGURE 3. The only difference from FIGURE 2 is the negative quantifier in the embedded VP which takes the position of the proper name *Nostalgia*. For reasons of space, information carried by identical tags as in FIGURE 2 is not repeated in FIGURE 3.

There are three negative quantifiers whose scope interaction must be determined. The restriction $[9] \in [13]$ (known from the previous example) leaves two possibilities: [9] could be identical with [6] or with [11]. If $[9] = [6]$ we are in the situation



in which the negative quantifier [6] of *niciun student* is interpreted in the embedded clause: Being identical with [0] it is a subterm of η and cannot take scope in the matrix clause. On top of this, the NEG CRITERION forces resumption between [6] and [11], we obtain a NC reading in the subjunctive clause and the interpretation (31a) for (26). If [0] = [11] the negative quantifier [6] can take scope in the matrix clause where it undergoes resumption with [7] to obey the NEG CRITERION. The result is a NC reading in the matrix clause and the interpretation (31b) for (26):

In this section we showed that our theory of NC in Romanian contains all basic ingredients to account for the properties of negative quantifiers and NC in complex sentences. The analysis is still incomplete in at least two respects: We did not properly integrate our theory of polyadic quantifiers with two-sorted type theory; and we did not carefully consider the full range of data that is relevant for a theory of NC in complex sentences. While the logical extension should be straightforward, the empirical questions are challenging. What are the speakers' intuitions about the scope of negative quantifiers in complex sentences with two or more *n*-words? An unconstrained theory predicts scope interactions that native speakers most likely will not perceive given the usual difficulties with multiple negations. It would be important to find out which readings are available and preferred, and

which grammatical or processing constraints are at play.

6 Conclusion

The present analysis of NC in Romanian applies the approach that was pioneered by an analysis of French in de Swart and Sag (2002). Our theory considerably extends de Swart and Sag's proposal by explicitly integrating a higher-order logic with polyadic quantification in HPSG. We expect that the formulation of the polyadic quantifier approach to NC in LRS will make it possible to unify this line of research with the typological approach to NC in Polish, French and German presented in Richter and Sailer (2006). Last but not least, adding polyadic quantification to LRS opens the door to exploring a whole range of new semantic phenomena in HPSG such as cumulative and *same/different* (unreducible) polyadic quantifiers (Keenan (1992), Keenan and Westerstahl (1997)). Since our constraint-based syntax-semantics interface supports the integration of polyadic quantifiers, HPSG theories can take full advantage of them. This brings within reach an explicit specification of the syntax and semantics of constructions that require unreducible polyadic quantifiers for an adequate rendering of their truth conditions and have, for that reason, turned out to be problematic in other grammar frameworks.

References

- Barbu, Ana-Maria. 2004. The negation NU: lexical or affixal item. In Emil Ionescu (ed.), *Understanding Romanian Negation. Syntactic and Semantic Approaches in a Declarative Perspective*, pages 68–82, Bucharest University Press.
- Ionescu, Emil. 1999. A Quantification-based Approach to Negative Concord in Romanian. In Geert-Jan M. Kruijff and Richard T. Oehrle (eds.), *Proceedings of Formal Grammar 1999*, pages 25–35, Utrecht.
- Ionescu, Emil. 2004. The Semantic Status of Romanian N-words in Negative Concord Constructions. In Emil Ionescu (ed.), *Understanding Romanian Negation. Syntactic and Semantic Approaches in a Declarative Perspective*, pages 83–118, Bucharest University Press.
- Iordăchioaia, Gianina. 2009. *Negative Concord with Negative Quantifiers: A Polyadic Quantifier Approach to Romanian Negative Concord*. Ph.D. thesis, submitted, University of Tübingen.
- Keenan, Edward L. 1992. Beyond the Frege Boundary. *Linguistics and Philosophy* 15, 199–221.
- Keenan, Edward L. and Westerstahl, Dag. 1997. Generalized Quantifiers in Linguistics and Logic. In Johan van Benthem and Alice ter Meulen (eds.), *Handbook of Language and Logic*, pages 837–893, Amsterdam: Elsevier Science.

- Klima, Edward. 1964. Negation in English. In Jerry A. Fodor and Jerrold J. Katz (eds.), *The Structure of Language*, pages 246–323, Englewood Cliffs: Prentice Hall.
- Ladusaw, William A. 1992. Expressing negation. In *Proceedings of SALT 2*, pages 237–259, Columbus: The Ohio State University.
- Lindström, Per. 1966. First Order Predicate Logic with Generalized Quantifiers. *Theoria* 32, 186–195.
- Peters, Stanley and Westerståhl, Dag. 2006. *Quantifiers in Language and Logic*. Oxford and New York: Oxford University Press.
- Przepiórkowski, Adam and Kupść, Anna. 1997. Negative Concord in Polish. Technical Report, Institute of Computer Science, Polish Academy of Sciences.
- Przepiórkowski, Adam and Kupść, Anna. 1999. Eventuality Negation and Negative Concord in Polish and Italian. In Robert Borsley and Adam Przepiórkowski (eds.), *Slavic in HPSG*, pages 211–246, Stanford: CSLI Publications.
- Richter, Frank and Kallmeyer, Laura. 2009. Feature Logic-based Semantic Composition: A Comparison between LRS and LTAG. In Anders Søgaard and Petter Haugereid (eds.), *Typed Feature Structure Grammars*, pages 32–92, Frankfurt a.M.: Peter Lang.
- Richter, Frank and Sailer, Manfred. 1999. LF Conditions on Expressions of Ty2: An HPSG Analysis of Negative Concord in Polish. In Robert Borsley and Adam Przepiórkowski (eds.), *Slavic in HPSG*, pages 247–282, Stanford: CSLI Publications.
- Richter, Frank and Sailer, Manfred. 2004. Basic Concepts of Lexical Resource Semantics. In *ESSLLI 2003 – Course Material I*, volume 5 of *Collegium Logicum*, Kurt Gödel Society Wien.
- Richter, Frank and Sailer, Manfred. 2006. Modeling Typological Markedness in Semantics: The Case of Negative Concord. In Stefan Müller (ed.), *Proceedings of the 13th International Conference on Head-Driven Phrase Structure Grammar*, pages 305–325, CSLI Publications.
- de Swart, Henriëtte and Sag, Ivan A. 2002. Negation and Negative Concord in Romance. *Linguistics and Philosophy* 25, 373–417.
- van Benthem, Johan. 1989. Polyadic Quantifiers. *Linguistics and Philosophy* 12, 437–464.
- Zeijlstra, Hedde H. 2004. *Sentential Negation and Negative Concord*. Utrecht: LOT Publications.