University of Lisbon

Department of Informatics

A Computational Grammar for Deep Linguistic Processing of Portuguese: LXGram, version 5

Francisco Costa António Branco

April 2014

Contents

List of Figures

List of Tables

Acknowledgments

We are very grateful for all the support and encouragement from Hans Uszkoreit, both during the phase of project proposal definition and during the development phase of the GramaXing project.

We are very grateful for all the invaluable help and for the conditions made available at the University of Saarland, Department of Computational Linguistics at Saarbruecken, that permitted the internship of Francisco Costa for 7 months, in 2005-2006.

We want to thank also Frederik Fouvry, who helped in the setting up of the project proposal, and to Valia Kordoni for the warm hospitality and support at Saarbrücken.

Part of the work in the grammar was also supported by the Ph.D. grant with the reference “SFRH / BD / 40140 / 2007”, by Fundação para a Ciência e Tecnologia.

The work on the grammar was also partly supported by the QTLeap project (project reference PTDC/EEI-SII/1940/2012).

Chapter 1

Preamble

This document is the report on the implementation of LXGram, in its version 5 of March 2014.

LXGram is a grammar for the computational processing of Portuguese.

It is being developed with the following major design goals:

- precision: it is a precision grammar delivering accurate, linguistically grounded information of natural language sentences
- deep processing: it is a grammar for deep linguistic processing in as much as besides information on the major syntactic dimensions of grammatical constituency and dependency, it delivers (and generates from) fully-fledged logical representation of the meaning of natural language sentences.
- large-scale: it is planned not to leave out any sort of regular grammatical construction or phenomena.
- multi-purpose: it is intended to make available as much linguistic information as it can possible be made explicit by automatic means, given the current state of the art in language technology, with the goal of offering itself to support the largest possible range of language technology applications.

LXGram is being developed at the University of Lisbon, by NLX—Natural Language and Speech Group of the Department of Informatics, Faculty of Sciences, under the coordination of António Branco. Major coding work has been performed by Francisco Costa. The development activities benefited from support or contributions from Mariana Avelãs, Filipe Gil, Marcos Gonzalez, Clara Pinto and David Raposo, Sérgio Castro, João Silva, Catarina Carvalheiro, Sílvia Pereira, Rita Santos, Juliana Franco and Will Roberts.

The development of LXGram has been undertaken with the support and in the scope of the DELPH-IN international consortium.

The research and development activities of LXGram were partially supported by FCT—Fundação para a Ciência e Tecnologia, of the Portuguese Ministry of Science, and to Instituto Camões, of the Portuguese Ministry of Foreign Affaires, under the research grant PLUS/PLP/50301/2003 for the project GramaXing. The former institution is also acknowledged for its partial support to the development of LXGram under the research grant RIPD/PSI/63557/2005 for the project SemanticShare.

For releases and licensing consult `http://nlx.di.fc.ul.pt/lxgram`.

Chapter 2

Introduction

2.1 Formalisms and Tools

LXGram is developed under the grammatical framework of Head-Driven Phrase Structure Grammar (HPSG: [Pollard and Sag, 1987], [Pollard and Sag, 1994], [Sag et al., 2003]) and uses Minimal Recursion Semantics (MRS: [Copestake et al., 2005]) for the representation of meaning.

This grammar implementation is undertaken with the LKB ([Copestake, 2002]) grammar development environment and its evaluation and regression testing is done via [incr tsdb()] ([Oepen, 2001]). It is also intended to be compatible with the PET parser ([Callmeier, 2000]).

The LinGO Grammar Matrix (version 0.9 [1]; [Bender et al., 2002]), an open-source kit for the rapid development of grammars based on HPSG and MRS, was used as the initial infrastructure upon which to build LXGram.

2.2 Implementation Agenda

The implementation of LXGram initially followed a list of phenomena aimed at being included in the grammar coverage. This implementation agenda was the following:

Phase A - Structure

1. Auxiliaries and basic phrase structure: S, VPs, PPs, APs, AdvPs
2. NPs (without relatives)
3. Predication structure and agreement
4. Modification structure

Phase B - Core Recursion

5. Completive subordination
6. Adverbial subordination
7. Punctuation
8. Coordination
9. Comparatives

Phase C - Reshuffling

10. Alternations: passive, middle, ...

[1] Initially, version 0.8 was used; at a later moment the coordination module provided in `http://depts.washington.edu/uwcl/HPSG2005/modules.html` was explored and then kept.

11. Raising
12. Control
13. Clitics I

Phase D - Long-distance Dependencies

14. Non-canonical word-order
15. Relatives
16. Interrogatives
17. Exclamatives and Imperatives

Phase E - Advanced Semantics

18. Negation
19. Clitics II
20. Tense and aspect
21. Determination and quantification

In version 5, each of these phenomena is covered in the grammar's implementation to some extent (with the exception of exclamative sentences), but not all of them are implemented with the same depth and comprehensiveness.

Chapter 3

Getting Started

The main files of LXGram are the following:

- Binaries:
  - `portuguese.grm` is the grammar compiled with `flop`.
- Configuration files:
  - All files in the `lkb` directory
    Configuration files for the LKB. Loading the file `script` in the LKB loads the grammar.
  - `portuguese.tdl` and all files in the `pet` directory
    Configuration files for the PET parser.
- Preprocessor definitions:
  - `preprocessor.fsr`
    Preprocessor rules (e.g. to undo contractions): these are only used when parsing from within the LKB. They are not used with PET and they are not used when treebanking sentences previously parsed with PET.
  - All files in the `chart-mapping` directory
    Type definitions for the chart mapping machinery
- Type definitions:
  - `matrix-redefined.tdl`
    Type definitions adapted from the LinGO Grammar Matrix.
  - `features-types.tdl`
    Definitions of many types and features.
  - `head-types.tdl`
    Definitions of the type hierarchy of types appropriate for the attribute HEAD. It is not the `head-types.tdl` file that comes with the LinGO Grammar Matrix.
  - `lexical-types.tdl`
    Definitions of lexical types.
  - `syntax.tdl`
    Definitions of syntactic rules.
  - `lexical-rules.tdl`
    Definitions of non-spelling-changing lexical rules.

  - `morphology.tdl`
    Definitions of supertypes of spelling-changing lexical rules.
  - `roots.tdl`
    Definition of start symbols.
  - `punctuation.tdl`
    Type definitions to control punctuation (see Section 5.9).

- Rules:
  - `rules.tdl`
    Names and inventory of syntactic rules.
  - `lrules.tdl`
    Names and inventory of non-spelling-changing lexical rules.
  - `irules.tdl`
    Names and inventory of spelling-changing lexical rules, as well as specification of those changes.
  - `irregs.tab`
    List of irregularities to spelling changes produced by the lexical rules.

- The lexicon:
  - All files in `lexicon2lkb`
    Code to convert .csv files into .tdl files containing lexical entries
  - `lexicon.src`
    The full set of manually created lexical entries
  - `lexicon.tdl`
    The manually created lexical entries that are loaded by the grammar
  - `lexicon/converted-lexicon.tdl`
    The lexical entries that are converted from the external lexicon and loaded by the grammar
  - Remaining files in the `lexicon` directory
    The scripts in this directory should be run by invoking the Makefile. They compare the manually created lexical entries (which are in `lexicon.src`) with the lexical entries converted from the external lexicon. It outputs the two files `lexicon.tdl` and `lexicon/converted-lexicon.tdl`, which are loaded by the grammar. Duplicate entries are output to only one of these files. In case of duplicates (a manual entry and the converted entry with the same lemma and a similar type), the script `lexicon/filter.py` controlls which one is kept.
  - `names/external-lexicon-names.tdl` This file includes lexical entries for all proper nouns found in the CINTIL annotated corpus. The remaining files in this directory contain the code used to automatically generate this file from that corpus.

- Additional files:
  - `labels.tdl`
    Definition of the labels associated with syntactic tree nodes.
  - `tree-decorations.tdl`
    Supertypes of the types in `labels.tdl`. These supertypes are used to factor out common behavior.

- `lkb/mtr.tdl`
  Definitions of types used in the generator and paraphraser rules.
- `trigger.mtr`
  Generator rules to select lexical items with no semantics in generation.
- `trigger-abstract-types.tdl`
  Supertypes of the types in `trigger.mtr`. These supertypes are used to factor out common behavior.
- `idioms.mtr`
  List of idiomatic expressions, used by the idiom detection mechanism (see Section 5.8).
- `idioms-abstract-types.tdl`
  Supertypes of the types in `idioms.mtr`. These supertypes are used to factor out common behavior.
- `paraphraser.mtr`
  Examples of rules that would enable the grammar to be used as a paraphraser (e.g. treating lexical synonymy).
- `Version.lsp`
  The grammar's name and version.
- All files in the `smaf` directory
  Configuration files for integrating the grammar with the shallow tools referred in Section 6.1, using SMAF. There are also several test examples in this directory.

[Copestake, 2002] is an introduction to the LKB system. The reader should consult it in order to know how to load and run an LKB grammar.

3.1 Patches

3.1.1 HCONS and `cheap`

There is a known issue affecting LXGram when it is used with the PET parser (`cheap`) in that the MRS representations often do not contain all the *qeq* constraints under the feature HCONS. This problem does not occur with the LKB. Because up to this point LXGram has been mostly used for treebanking, and in this process the semantic representations for sentences previously parsed by `cheap` are rebuilt in the LKB, this problem has not been fixed, as the resulting treebank is not affected by it.

3.1.2 `nil` predicate names

LXGram has been used in a setup where the sentences to be parsed by the grammar are first analyzed by a part-of-speech tagger and morphological analyzer named LX-Suite. Another tool, `lxsuite2fsc.jar` transforms the output of LX-Suite into XML in the FSC (Feature Structure Chart) format that `cheap` accepts as input.

The parser is run with the `-tsdbdump` option, which makes `cheap` write the parsing results in a format that can be read by [incr tsdb()]. This is then treebanked with [incr tsdb()]: a human annotator selects the best parse of each sentence, from the best few hundred parses delivered by the grammar (a limit of 250 parses has been in use). The resulting collection of parsed sentences forms a treebank. After treebanking, the result is exported to separate files. For each sentence, a file is created which contains the sentence and the selected analysis. This is performed with [incr tsdb()].

The use of LX-Suite allows LXGram to deal with unknown words (words not listed in its lexicon) by assigning them default lexical entries according to their part of speech (as recognized by LX-Suite). When the sentences are treebanked in [incr tsdb()], the information coming from LX-Suite is no longer available, and the semantic representations are rebuilt in the LKB. The name of the predicate (that appears in the semantic representations) for the unknown word is lost in this process. When the treebanked sentences are exported to files, instead of the predicate names what one finds is `nil`.

There is currently a tool, `lkbnils.sh`, which corrects the exported files, replacing occurrences of `nil` with the appropriate predicate name. In order to determine these predicate names, the tool also looks at the files produced by `lxsuite2fsc.jar`.

3.1.3 Semantic discriminants and multiple remaining analysis

During treebanking with LXGram, the option of using semantic discriminants has been chosen. That is, when a human annotator selects the best parse among the parses returned by LXGram, the choices the human annotator must make are based on the semantic representations associated with the parses. The alternative would consist in choosing the syntax rules used in each analysis and their relative scope in each analysis.

By using semantic discriminants, the annotator does not need to know about the specific syntax rules used by the grammar. There is however a small number of sentences for which the grammar produces different parses with similar semantics. These cases cannot be distinguished on the basis of semantics (the semantics is the same for two or more parses), and therefore it is not possible to select a unique parse. Some of these cases are unavoidable. For instance, if a grammar can produce both SVO and OVS analyses for a sentence such as *Dog eats dog* (these two possibilities are appropriate in the case of Portuguese), the semantic discriminants machinery will not be able to distinguish between these two analyses: in both analyses, the first argument of the barking relation is a dog, and its second argument is also a dog.

When treebanking with LXGram in the NLX group, it has been decided that the human annotators should accept these sentences if the topmost analysis in the [incr tsdb()] user interface is correct and natural from the syntactic point of view. Similarly, there is another tool, `export2exportWoutForest.py`, which processes all exported files (one file is generated for each sentence). For each sentence, if more than one analysis remains for it, only the first one is kept, and all others are deleted from the file.

Chapter 4

Linguistic Coverage and Performance

At this point, LXGram size can be described with the numbers presented in Table 4.1.

Grammar Version	Lines of Code	Types (with GLBs)	Syntactic Rules	Lexical Entries	Lexical Types (Leaves)	Lexical Rules	Maintenance	Progress
A.1.2	10,600	3,069	28	1,112	174	30		Phase A.1
A.4.1	24,484	7,787	53	2,718	409	40		Phase A.4
5	43,798	24,960	107	26,119	1,179	81		Phase E.21

Table 4.1: LXGram implementation progress.

The remaining tables describe LXGram's coverage, overgeneration and performance on two disjoint test suites: the test suite for the phase A.1 (see Section 2.2), containing 202 examples, and the test suite for the phases A.2, A.3 and A.4, with 851 items. These test suites are hand-crafted and contain many examples of the relevant phenomena, as well as many negative examples, that are to be rejected by the grammar.

We present the results for the version A.1.2 of the grammar (corresponding to the end of Phase A.1) and for the version A.4.1 of LXGram (at the end of phase A.4), for comparative purposes. The tables presenting these results are adapted from [incr tsdb()] output (the leftmost columns were removed and the table titles abridged).

Table 4.2 describes the coverage of LXGram on the data set for phase A.1, both under version A.1.2 and under version A.4.1. Coverage was maintained from version A.1.2 to version A.4.1. Ambiguity rose slightly (from an average of 2.48 parses per sentence to 2.93), increasing by 18%. Lexical ambiguity increased by 54% (from 59.11 to 90.84). Table 4.3 presents the coverage of version A.4.1 on the test suite for the phase A.4.

Table 4.4 and Table 4.5 display the values for overgeneration. The version A.4.1 is overgenerating for the phase A.1 test suite. The main source of overgeneration is sequences that are impossible as an adjective phrase receiving a parse as a noun phrase with a missing noun. One example is the following, where "mais bom" is intended to mean "melhor" (*better*):

Phase A.1 Test Suite
LXGram Version A.1.2

Coverage						
total	**positive**	**word**	**lexical**	**distinct**	**total**	**overall**
items	**items**	**string**	**items**	**analyses**	**results**	**coverage**
♯	♯	ϕ	ϕ	ϕ	♯	%
202	**122**	**6.86**	**59.11**	**2.48**	**122**	**100.0**

Phase A.1 Test Suite
LXGram Version A.4.1

Coverage						
total	**positive**	**word**	**lexical**	**distinct**	**total**	**overall**
items	**items**	**string**	**items**	**analyses**	**results**	**coverage**
♯	♯	ϕ	ϕ	ϕ	♯	%
202	**122**	**6.86**	**90.84**	**2.93**	**122**	**100.0**

Table 4.2: LXGram coverage evolution on the phase A.1 test suite.

Phase A.4 Test Suite
LXGram Version A.4.1

Coverage						
total	**positive**	**word**	**lexical**	**distinct**	**total**	**overall**
items	**items**	**string**	**items**	**analyses**	**results**	**coverage**
♯	♯	ϕ	ϕ	ϕ	♯	%
851	**422**	**5.31**	**33.55**	**1.15**	**422**	**100.0**

Table 4.3: LXGram coverage evolution on the phase A.4 test suite.

(1) * Este computador é mais bom.
this computer is more good

Although the expression "mais bom" is ungrammatical with "mais" as an adverb, it can receive a parse as an NP (with "mais" as a determiner), because of examples like:

(2) Queremos menos vinho mau e mais bom.
we want less wine bad and more good
We want less bad wine and more of the good one.

With the version A.1.2 of the grammar, sentences like (1) were ruled out. The version A.4.1 contains an implementation of noun ellipsis (see Section 8.7) that produces a parse for these sentences. However, phrases like "mais bom" keep not being analyzed as adjective phrases.

Efficiency measures are also provided in Table 4.6 and Table 4.7. In Table 4.6 we can see that the average time needed to analyze a sentence in the phase A.1 test suite went up from 0.06 seconds in version A.1.2 to 0.22 seconds in version A.4.1. The amount of memory required also increased from 14.2MB to 27.5MB. These numbers reflect the higher number of rules in the version A.4.1 of LXGram as well as the increased lexical ambiguity.

Phase A.1 Test Suite						
LXGram Version A.1.2						
Overgeneration						
total items ♯	**negative items** ♯	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** ♯	**overall coverage** %
202	**80**	**8.90**	**82.59**	**0.00**	**0**	**0.0**

Phase A.1 Test Suite						
LXGram Version A.4.1						
Overgeneration						
total items ♯	**negative items** ♯	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** ♯	**overall coverage** %
202	**80**	**8.90**	**128.00**	**1.14**	**14**	**17.5**

Table 4.4: LXGram overgeneration evolution on the phase A.1 test suite.

Phase A.4 Test Suite						
LXGram Version A.4.1						
Overgeneration						
total items ♯	**negative items** ♯	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** ♯	**overall coverage** %
851	**429**	**5.38**	**33.74**	**0.00**	**0**	**0.0**

Table 4.5: LXGram overgeneration evolution on the phase A.4 test suite.

4.1 Version 5

During the development of version 5 of LXGram, 1000 sentences taken from the CINTIL corpus were used for regression testing. These correspond to the sentences in the testsuites SSTreeB-bCINT-040 to SSTreeB-bCINT-059 of the CINTIL Treebank. Since this data set does not include negative examples, it does not make sense to measure overgeneration.

The previous version of the grammar, version A.4, showed the coverage results presented in Table 4.8. On the same data set, version 5 shows the coverage in Table 4.9. For this specific data set, coverage increased from 16.9% to 35.7%.

The values for efficiency are in Table 4.10 for version A.4.1 and in Table 4.11 for version 5, using the same data. By comparing these two tables, it can be seen that the amount of time required to parse a sentence has gone up very considerably. The amount of memory required has also increased, but less markedly.

Phase A.1 Test Suite						
LXGram Version A.1.2						
Performance						
items $\sharp$	**etasks** ϕ	**filter** %	**edges** ϕ	**first** ϕ (s)	**total** ϕ (s)	**space** ϕ (kb)
202	**282**	**92.5**	**172**	**0.07**	**0.06**	**14516**

Phase A.1 Test Suite						
LXGram Version A.4.1						
Performance						
items $\sharp$	**etasks** ϕ	**filter** %	**edges** ϕ	**first** ϕ (s)	**total** ϕ (s)	**space** ϕ (kb)
202	**1374**	**95.1**	**314**	**0.23**	**0.22**	**28136**

Table 4.6: LXGram efficiency evolution on the phase A.1 test suite.

Phase A.4 Test Suite						
LXGram Version A.4.1						
Performance						
items $\sharp$	**etasks** ϕ	**filter** %	**edges** ϕ	**first** ϕ (s)	**total** ϕ (s)	**space** ϕ (kb)
848	**733**	**95.0**	**167**	**0.12**	**0.11**	**15843**

Table 4.7: LXGram efficiency evolution on the phase A.4 test suite.

CINTIL Sentences: Testsuites SSTreeB-bCINT-040 to SSTreeB-bCINT-059							
LXGram Version A.4.1							
Coverage							
Phenomenon	**total items** $\sharp$	**positive items** $\sharp$	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** $\sharp$	**overall coverage** %
Total	**1000**	**1000**	**26.91**	**61.44**	**1.00**	**169**	**16.9**

Table 4.8: LXGram (version A.4) coverage on CINTIL sample

CINTIL Sentences: Testsuites SSTreeB-bCINT-040 to SSTreeB-bCINT-059							
LXGram Version 5							
Coverage							
Phenomenon	**total items** $\sharp$	**positive items** $\sharp$	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** $\sharp$	**overall coverage** %
Total	**1000**	**1000**	**25.75**	**867.72**	**1.00**	**357**	**35.7**

Table 4.9: LXGram (version 5) coverage on CINTIL sample

CINTIL Sentences: Testsuites SSTreeB-bCINT-040 to SSTreeB-bCINT-059							
LXGram Version A.4.1							
Performance							
Phenomenon	**items**	**etasks**	**filter**	**edges**	**first**	**total**	**space**
	$\sharp$	ϕ	%	ϕ	ϕ (s)	ϕ (s)	ϕ (kb)
Total	**360**	**190126**	**98.5**	**54966**	**1.03**	**5.39**	**264658**

Table 4.10: LXGram (version A.4) efficiency on CINTIL sample

CINTIL Sentences: Testsuites SSTreeB-bCINT-040 to SSTreeB-bCINT-059							
LXGram Version 5							
Performance							
Phenomenon	**items**	**etasks**	**filter**	**edges**	**first**	**total**	**space**
	$\sharp$	ϕ	%	ϕ	ϕ (s)	ϕ (s)	ϕ (kb)
Total	**637**	**156099**	**98.6**	**24064**	**17.03**	**21.18**	**356615**

Table 4.11: LXGram (version 5) efficiency on CINTIL sample

Chapter 5

General Aspects of the Implementation

In the present chapter some of the strategies followed in the implementation of LXGram are described. For instance, there is often a concern to reduce feature structures (Section 5.1). Section 5.2 and Section 5.3 describe departures from the LinGO Grammar Matrix. The remaining sections describe some aspects of the implementation that have an impact on the general organization of the features employed in the grammar.

5.1 Reduction of Feature Structures via Type Inference and Polymorphism

Throughout the implementation of LXGram a technique described in [Flickinger, 2000] and referred to as *minimal types* is used in order to reduce the size of the feature structures that the parser will operate on at runtime. The use of minimal types is employed in the ERG and in the LinGO Grammar Matrix.

An example follows. The features SLASH, REL and QUE are needed under NON-LOCAL, so an initial approach could be to create a type *non-local* for which these features are declared appropriate and, since NON-LOCAL is a feature of synsems, declare the type *synsem* to have a feature NON-LOCAL of type *non-local*.

The minimal types approach would be different. One still has type *non-local*, where all these features are declared, but one would also create an additional type *non-local-min*, which has no features, and declare *non-local* to inherit from *non-local-min*. Furthermore, the feature NON-LOCAL would now be declared in *synsem* to be of type *non-local-min*. The following figures show the contrast between not using (Figure 5.1) and using (Figure 5.2 and Figure 5.3) the *minimal types* approach.

The LKB supports polymorphism, so the feature NON-LOCAL can be instantiated with a feature structure of the type *non-local*, since it is a subtype of *non-local-min*.

$$\begin{bmatrix} \textit{synsem} & \\ \textsc{local} & \textit{local-min} \\ \textsc{non-local} & \begin{bmatrix} \textit{non-local} & \\ \textsc{slash} & \textit{0-1-dlist} \\ \textsc{que} & \textit{0-1-dlist} \\ \textsc{rel} & \textit{0-1-dlist} \end{bmatrix} \end{bmatrix}$$

Figure 5.1: *Synsem* without a minimal type for attribute NON-LOCAL

$$\begin{bmatrix} \textit{synsem} & \\ \textsc{local} & \textit{local-min} \\ \textsc{non-local} & \textit{non-local-min} \end{bmatrix}$$

Figure 5.2: *Synsem* with a minimal type for attribute NON-LOCAL

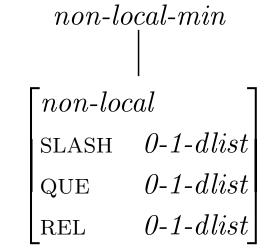


Figure 5.3: Example type hierarchy under *non-local-min*.

Furthermore, the LKB requires for each feature there to be a single most general type for which it is appropriate; if such a constraint is not respected there is an error at grammar load time. This requirement enables the system to support type inference. With type inference, one does not need to mention explicitly that a particular instance of the feature NON-LOCAL is of the type *non-local* when one wants to use these features. Whenever a constraint mentions the feature SLASH, QUE or REL, the type *non-local* is inferred for that instance of the feature NON-LOCAL.

The main advantage of using minimal types is that it can make feature structures smaller. Continuing with the NON-LOCAL example, the features SLASH, QUE and REL will appear in feature structures only if they are mentioned in some constraint (e.g. in general they will not appear in synsems in a MOD or SPEC list, as these are not used in the amalgamation of non-local features).

This fact will have a positive impact on performance, as it spares some unification operations.

A nice side effect is that, since features that are not being used are not present in the resulting AVMs, these become more readable as they are smaller and redundant information is hidden.

As noted above, the ERG and the LinGO Grammar Matrix use this technique. In fact, the types *non-local-min* and *non-local* are defined in both of them in a way similar to the one presented.

The PET parser manipulates feature structures with the purpose of removing redundant features automatically at run-time (*unfilling*, see [Callmeier, 2000]), which is in fact equivalent to using minimal types in the grammar. However, [Flickinger, 2000] reports that the use of minimal types in a grammar can still improve parser performance even when the grammar is used with a parser that *unfills* feature structures.

In LXGram, this technique is also employed. Furthermore, instead of creating a single abstract type and a more specific one where all the features that are needed are introduced at once, sometimes an extra abstract type for each feature is created (where that feature is introduced), and all possible combinations of subtypes of these are also created.

Therefore, for each feature in a feature structure, it will be present there if and only if some constraint mentions it, and since there is a subtype for each combination of features, the technique will never cause unification failures by itself, thus having no impact on correctness.

This technique can be illustrated with the type hierarchy under *png*, where the features PERSON, NUMBER and GENDER are necessary (Figure 5.4). These features are the place where the information about morphological gender, number and person is represented in the grammar. For instance, the type hierarchy presented allows for a *png* object underspecified for gender not to have an attribute GENDER, instead of having one with its most general type *gender*.

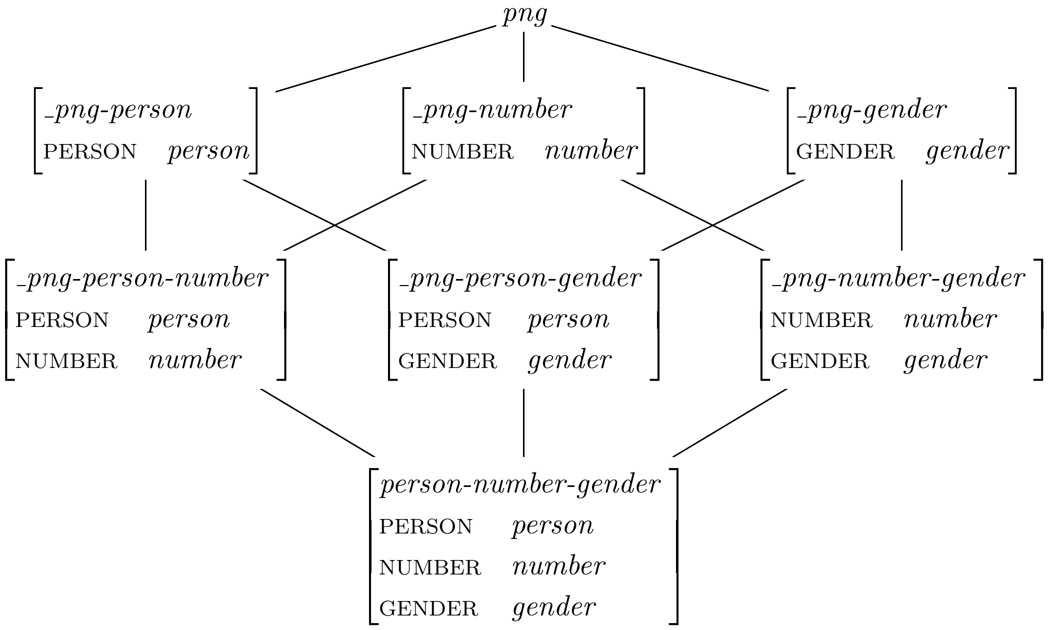


Figure 5.4: Type hierarchy under *png* with expanded constraints. Features PERSON, NUMBER and GENDER are declared in the types *_png-person*, *_png-number* and *_png-gender* respectively.

5.2 Changes to the LinGO Grammar Matrix

The initial policy in the implementation of LXGram was to not change the code in the LinGO Grammar Matrix. As the grammar grew, however, certain changes happened to be required.

The main motivation for not changing the Matrix setup is that it makes it easier to upgrade the grammar to new versions of the Matrix. Also, not changing it means that it is guaranteed that the construction of semantics is done as in the other grammars that use the LinGO Grammar Matrix, which helps keep semantic representations consistent and uniform across grammars.

LXGram is growing faster than the Matrix is being updated, so the first advantage has had no practical consequences so far, and one can also make semantic representations similar to the ones created by Matrix grammars without having to use the Matrix unchanged.

As the changes accumulated over time, we decided to abandon the original `matrix.tdl` file and include the changed type definitions in a new file, called `matrix-redefined.tdl`.

The next section documents an important departure from the LinGO Grammar Matrix.

5.3 Functors

In LXGram we replace the Head-Specifier and Head-Adjunct configurations with the Functor-Head schemata of [Allegranza, 1998a], [Allegranza, 1998b], [Van Eynde, 2003a] and [Van Eynde, 2003b].

In [Allegranza, 1998a], [Allegranza, 1998b], [Van Eynde, 2003a] and [Van Eynde, 2003b], Head-Functor relations are a cover of Head-Specifier and Head-Adjunct configurations. Functors, like adjuncts, select their head via a dedicated feature. All treatments put this feature that encodes selection requirements under the attribute HEAD, but the name varies (here it is SELECT). As a consequence, this feature SELECT percolates in all headed constructions. Like Head-Specifier configurations, information about saturation of the resulting node (i.e. its combinatorial potential) may be different from the combinatorial potential of the head daughter. So, for instance, a noun can combine with a determiner on the left to form a phrase with a determiner and noun, and this phrase cannot combine with another determiner on the left.

In Head-Specifier configurations, information about this kind of combinatorial potential is determined by the head noun, not by the specifier. Nouns select their specifier in their SPR attribute, which is list-valued (and usually either the empty list or a singleton list). Head-Specifier constructions unify the synsem in that attribute with the synsem of the other daughter, and the SPR of the mother node is the tail of the SPR list in the head daughter.

In Head-Functor schemata, saturation of the mother node is determined by the functor daughter, not the head (details are below).

With functors replacing specifiers and adjuncts, the features MOD, SPEC and SPR are no longer necessary, and neither are Head-Specifier phrases and Head-Adjunct phrases.

LXGram uses the features SELECT, MARKING and MARK to implement Head-Functor schemata. MARKING is used to describe combinatorial potential other than saturation of a head's arguments. For instance, since cardinals cannot iterate, the top node of a phrase like *three cars* would have a different value of MARKING from the one of *cars*.

The features MARK and SELECT are relevant for functors. The value of MARKING of the mother node in Head-Functor phrases comes from the MARK attribute of the functor daughter. The attribute SELECT is where functors state the heads they can attach to.

The main properties of Head-Functor schemata are presented in Figure 5.5.

$$\begin{bmatrix} \textit{basic-head-functor-phrase} \\ \text{SYNSEM}|\text{LOCAL}|\text{CAT} \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{VAL}|\text{SUBJ} & \boxed{2} \\ \text{MARKING} & \boxed{3} \end{bmatrix} \\ \text{HEAD-DTR}|\text{SYNSEM } \boxed{4} \begin{bmatrix} \text{LOCAL}|\text{CAT} \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{VAL}|\text{SUBJ} & \boxed{2} \end{bmatrix} \end{bmatrix} \\ \text{NON-HEAD-DTR}|\text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{HEAD}|\text{MARKER} \begin{bmatrix} \text{MARK} & \boxed{3} \\ \text{SELECT} & \boxed{4} \end{bmatrix} \end{bmatrix}$$

Figure 5.5: Outline of Head-Functor schemata

It is a headed construction, so the HEAD of the mother is token-identical to the HEAD of the head daughter. This constraint — the Head Feature Principle — is inherited from a supertype coming from the LinGO Grammar Matrix, *headed-phrase*, which implements this principle.

The subject of the mother and the subject of the head daughter are also shared since functors do not discharge the subject of the head. In LXGram the COMPS feature of the mother node can be the COMPS of the head daughter or the COMPS of the non-head daughter (see below).

Because functors have access to information about valence and marking of the head (the functor daughter's SELECT feature is unified with the head daughter's SYNSEM), control on the level of saturation of the head in these constructions is reduced to lexical specifications in functors.

Functors can cause the saturation described by the feature MARKING on the mother node to be different from the one on the head daughter — the mother node's MARKING feature is structure-shared with the functor's MARK feature.

LXGram contains three functor-head rules inheriting from *basic-head-functor-phrase*. One projects a functor to the right of the head (*head-functor-phrase*), which identifies the COMPS attribute of the resulting phrase with that of the head daughter. The other two project a functor to the left of its head. One of these identifies the complements of the mother with those of the head daughter (*functor-head-hcomps-phrase*), the other percolates the complements of the functor (*functor-head-fcomps-phrase*). Sentence (3), where the "(do) que" (*than*) phrase is analyzed as a complement of "mais" (*more*), illustrates the need for these three rules. A partial parse tree

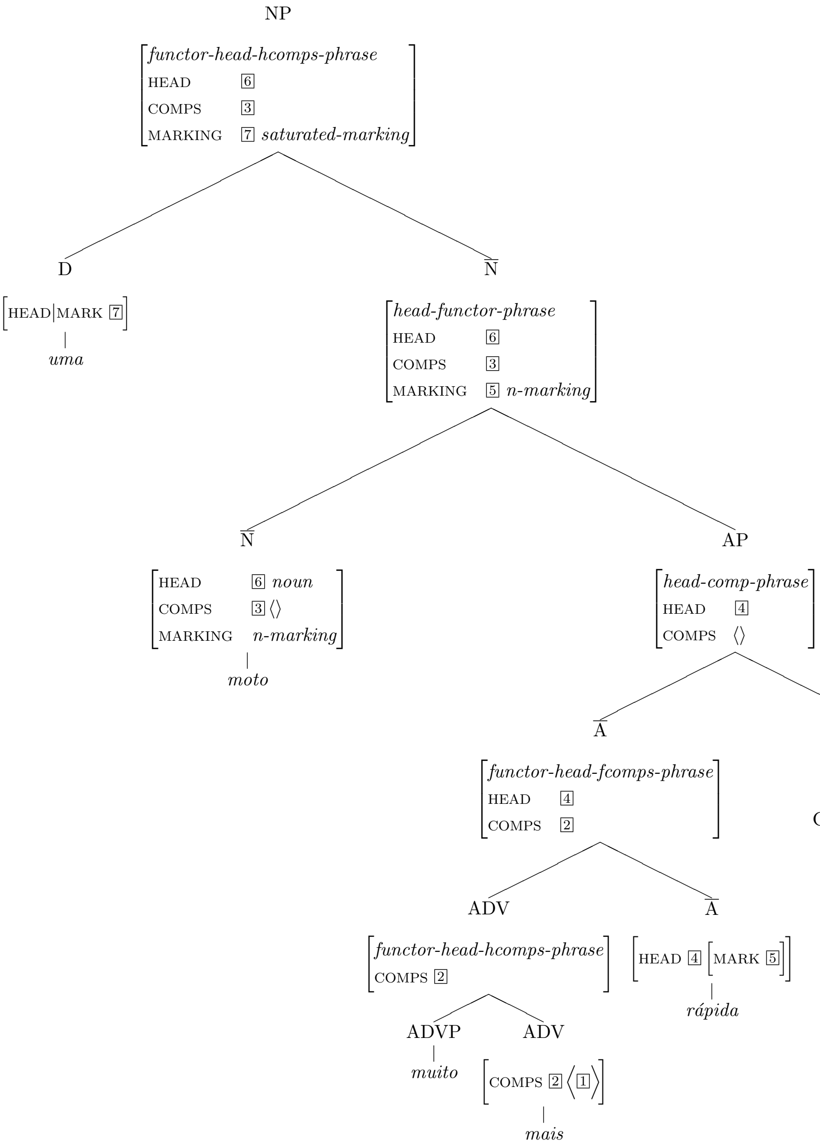


Figure 5.6: Parse tree for a phrase with all functor-head rules. The phrase is "uma moto muito mais rápida que esta" (*a motorcycle much faster than this one*). Feature paths are shortened.

is presented in Figure 5.6, where each node is decorated with the corresponding rule (the values that are presented here for the attribute MARKING are explained in Chapter 8).

(3) Tenho uma moto muito mais rápida (do) que esta.
I have a mortorcycle much more fast than this one
I have a motorcycle much faster than this one.

This example also illustrates the purpose of the features MARKING and MARK. What distinguishes the NP node from the $\overline{\mathrm{N}}$ nodes in this example is the value of the feature MARKING, which comes from the MARK feature of the functor daughter.

Also, there are actually two versions for each of these three rules (so a total of six rules): a version for intersective modifiers, another version for scopal modifiers and specifiers. The purpose of this splitting is explained in the next subsection.

5.3.1 Composition of Semantics in Head-Functor Phrases

The composition of semantics comes as expected: as all semantic information comes from the daughters, the RELS and HCONS of the mother are simply the difference list append of the homonymous features of both daughters.

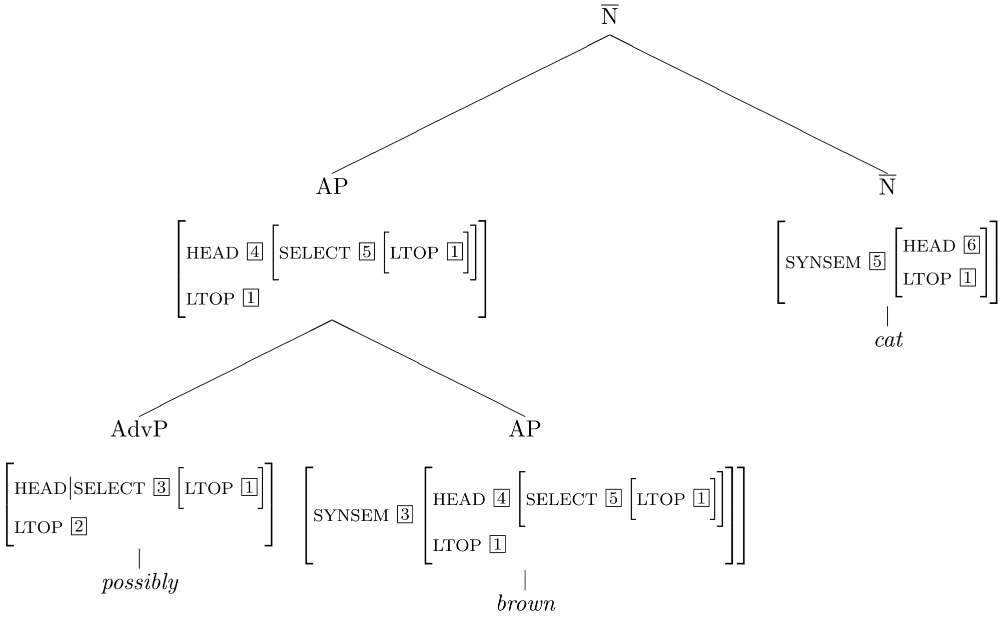
As discussed elsewhere [Kasper, 1996], [Copestake et al., 2005], there are issues (concerning the feature LTOP) regarding the interaction between intersective and scopal modifiers. The next paragraphs describe the problem at hand.

Consider an example like *possibly brown cat*, where the adjective *brown* is an intersective modifier of the noun *cat*, and the adverb *possibly* is a scopal modifier of the adjective *brown*.

Assume that intersective modifiers unify their LTOP feature with the LTOP of the synsem they select via their SELECT attribute in their lexical entries, so that the relations for the modifying element and the modified one end up with the same LBL in the MRS (as this situation denotes conjunction of the two relations). Note that in general the LTOP of a lexical entry will be unified with the LBL of a relation in that item's RELS, so unifying LTOPs amounts to unifying LBLs in MRSs.

As for scopal modifiers (as well as determiners, which now, as functors, combine via the same rules), they do not identify these values but rather use the LTOP of the selected constituent as the value for one of the arguments of the relation they introduce, often mediated by a *qeq* constraint in the functor's HCONS — and all of this is done in the lexical entries for functors.

This yields the wrong semantics for phrases like *possibly brown cat*. What is produced is equivalent to $\lambda x.possible(brown(x) \wedge cat(x))$, but what is correct is $\lambda x.possible(brown(x)) \wedge cat(x)$ (feature paths are shortened in the derivation tree):


N̄
AP
N̄
HEAD 4 [SELECT 5 [LTOP 1]]
LTOP 1
SYNSEM 5 [HEAD 6, LTOP 1]
cat
AdvP
AP
HEAD|SELECT 3 [LTOP 1]
LTOP 2
possibly
SYNSEM 3 [HEAD 4 [SELECT 5 [LTOP 1]], LTOP 1]
brown

As can be seen, because LTOPs are unified in the lexicon, semantic scope and syntactic scope do not match.

There are two solutions in the literature. One is in [Copestake et al., 2005]: LTOP features are not unified in the lexicon, but in the syntax rules. This requires separate rules for intersective modification (where LTOP attributes are unified) and scopal modification (where these features are not unified). The other is in [Kasper, 1996]: in order to obtain the desired result, the number of features used for the composition of semantics is enriched.

In LXGram we follow the first solution. Rule application is controlled by a boolean feature under MARKER, called SCOPAL. Scopal modifiers come in the lexicon with the value + for this

$$\begin{bmatrix} \textit{head-initial} & \\ \text{HEAD-DTR} & \boxed{1} \\ \text{NON-HEAD-DTR} & \boxed{2} \\ \text{ARGS} & \left\langle \boxed{1}, \boxed{2} \right\rangle \end{bmatrix} \begin{bmatrix} \textit{head-final} & \\ \text{HEAD-DTR} & \boxed{1} \\ \text{NON-HEAD-DTR} & \boxed{2} \\ \text{ARGS} & \left\langle \boxed{2}, \boxed{1} \right\rangle \end{bmatrix}$$

Figure 5.7: Implementation of word order in binary headed phrases

feature, and intersective modifiers have the type - here. A set of Head-Functor rules requires functor daughters with a SCOPAL feature of type - and identifies the LTOP feature of both daughters. Another set of rules constrains the functor daughter with the other value for this feature and places no constraints on the LTOP features.

5.3.2 Word Order in Head-Functor Phrases

In the LKB, the actual daughters of a rule are configured with the parameter ***args-path***. Its value is usually ARGS, a feature of *sign* instances.[1] It must be list-valued (it is a list of *sign*s), and the position of an element in that list correlates with linear precedence. In many computational grammars and in the LinGO Grammar Matrix, features like the attribute HEAD-DTR used in Figure 5.5 can be viewed as pointers to specific elements of that list.

It is often the case that abstract types for phrases are employed to constrain these pointer features, and then different subtypes implement different word order possibilities by linking them to different elements in ARGS. Figure 5.7 shows the constraints for two abstract types that define word order between the head daughter and the non-head daughter in binary headed phrases: *head-initial* constrains the head daughter to precede the non-head daughter, and *head-final* defines the non-head daughter to precede the head daughter. These types are in the LinGO Grammar Matrix.

The essential properties of Head-Functor phrases are stated in the abstract type *basic-head-functor-phrase*, already presented in Figure 5.5. Two subtypes implement the two word order possibilities between the daughters, by inheriting from *head-initial* or *head-final*. The result is shown in Figure 5.8. All headed phrases with two daughters inherit from *basic-binary-headed-phrase*.

As stated, all functors can feed both sets of rules. This situation is not desirable, since specific pairings of head and functor can be restricted to occur in a specific order.

For instance, a preposition comes in the lexicon with the information that it must attach to an NP on its right (forming a PP)[2]and then modify nouns and verbs. PPs can attach to either side of a verb headed constituent (4a, 4b), but only to the right of nouns (4c, 4d).

(4) a. Isso sai [$_{\text{PP}}$ com benzina.]
that goes away with benzine
That goes away with benzine.

b. Isso [$_{\text{PP}}$ com benzina] sai.
that with benzine goes away
That goes away with benzine.

[1] The feature ARGS is declared in the LinGO Grammar Matrix as appropriate for *signs*, but it does not make sense to speak of the ARGS of lexical items, so in LXGramARGS is declared in type *phrase-or-lexrule* instead. This type also comes in the LinGO Grammar Matrix. It is the supertype of all phrases and lexical rules (which for instance account for morphological inflection), but not of lexical items.

[2] This is achieved by constraining the element in the COMPS of prepositions to be of type *canonical-synsem*, as in Portuguese complements of prepositions cannot be null (hence they cannot unify with *unexpressed-synsem*), and they cannot be extracted, either (hence they must be incompatible with *gap*).

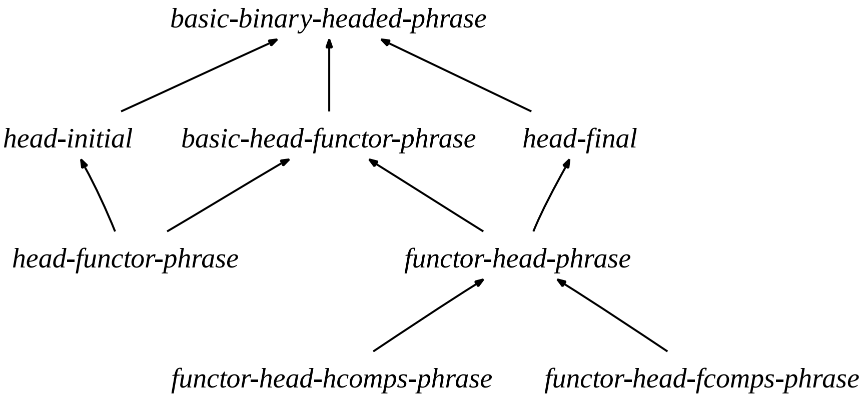

basic-binary-headed-phrase
head-initial
basic-head-functor-phrase
head-final
head-functor-phrase
functor-head-phrase
functor-head-hcomps-phrase
functor-head-fcomps-phrase

Figure 5.8: Organization of Head-Functor phrases.

c. Era um chapéu [PP com uma antena.]
was a hat with an antenna
It was a hat with an antenna.

d. * Era um [PP com uma antena] chapéu.
was a with an antenna hat

The way word order is controlled is by using more features to denote word order restrictions. These restrictions are seen as properties of functors, i.e. it is assumed that word order restrictions are lexical properties of functors.

The two features that are used are also under MARKER: PREHEAD and POSTHEAD. They contain constraints that must be satisfied when a functor precedes or follows the head daughter, respectively. These constraints are put on SELECT and MARK attributes under PREHEAD and POSTHEAD. The basic organization of features under MARKER is in Figure 5.9. The AVM in this figure does not correspond to a type definition. It is rather a schematic view of the features that are used.

$$\begin{bmatrix} \text{MARKER} & \begin{bmatrix} \text{SELECT} & \textit{synsem} \\ \text{MARK} & \textit{marking} \\ \text{PREHEAD} & \begin{bmatrix} \text{SELECT} & \textit{synsem} \\ \text{MARK} & \textit{marking} \end{bmatrix} \\ \text{POSTHEAD} & \begin{bmatrix} \text{SELECT} & \textit{synsem} \\ \text{MARK} & \textit{marking} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 5.9: Organization of the features under the HEAD of functors.

The head type for prepositions has the constraints in Figure 5.10, where *noun-or-verb* is a supertype of *verb* and *noun*.

In order for these constraints to play the intended role, the two types *head-functor-phrase* and *functor-head-phrase*, depicted in Figure 5.8, must be further refined. Their definitions are in Figure 5.11.

Because the higher SELECT attribute is already unified with the SYNSEM of the head daughter, and the higher MARK with the MARKING of the mother node, the homonymous features under PREHEAD and POSTHEAD will also be unified with these, but only when the relevant syntax rule is used.

$$\begin{bmatrix} \textit{preposition} & \\ \text{MARKER} & \begin{bmatrix} \text{SELECT|LOCAL|CAT|HEAD} & \textit{noun-or-verb} \\ \text{PREHEAD|SELECT|LOCAL|CAT|HEAD} & \textit{verb} \end{bmatrix} \end{bmatrix}$$

Figure 5.10: Constraints on the HEAD of prepositions. *noun-or-verb* is a supertype of *noun* and *verb*, the head types of nouns and verbs, respectively.

$$\begin{bmatrix} \textit{head-functor-phrase} & \\ \text{NON-HEAD-DTR|SYNSEM|LOCAL|CAT|HEAD|MARKER} & \begin{bmatrix} \textit{post-marker} & \\ \text{SELECT} & \boxed{1} \\ \text{MARK} & \boxed{2} \\ \text{POSTHEAD} & \begin{bmatrix} \text{SELECT} & \boxed{1} \\ \text{MARK} & \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{functor-head-phrase} & \\ \text{NON-HEAD-DTR|SYNSEM|LOCAL|CAT|HEAD|MARKER} & \begin{bmatrix} \textit{pre-marker} & \\ \text{SELECT} & \boxed{1} \\ \text{MARK} & \boxed{2} \\ \text{PREHEAD} & \begin{bmatrix} \text{SELECT} & \boxed{1} \\ \text{MARK} & \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 5.11: Constraints on *head-functor-phrase* and *functor-head-phrase*

In LXGram the type of the feature MARKER controls whether specific functors can feed head-initial and head-final Head-Functor constructions. An element with this feature constrained to be of the type *pre-only-marker* cannot feed the head-initial Head-Functor rule, because in this rule the functor daughter is constrained to have a feature MARKER with a type incompatible with *pre-only-marker*. This incompatible type is a type subsumed by *post-marker*. An element with the value *post-only-marker* for this feature is not elligible to be the functor daughter of head-final Head-Functor phrases, either, for similar reasons (the head-final Head-Functor phrases constrain the MARKER of their functor daughter to be of a type subsumed by *pre-marker*). The type hierarchy for these values is in Figure 5.12.

The hierarchy presented in this figure is simplified in that minimal types, as presented in Section 5.1, are also used in the grammar, but they are not shown in this figure.

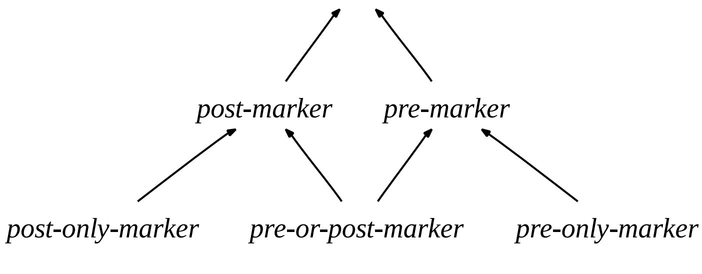


Figure 5.12: Values for the feature MARKER (simplified hierarchy).

5.4 Sentence Force

Like other DELPH-IN grammars, LXGram no longer produces message relations in MRSs. Instead, a feature SF (SENTENCE FORCE) in events holds information on illocutionary force. Declarative sentences correspond to MRSs where the main event has an SF feature with value *proposition*, interrogative sentences give rise to MRSs with an event with a *question* SF and so on.

At the begininng of 2007 a new version of the LinGO Grammar Matrix appeared where message relations had been replaced with this encoding of illocutionary force. This change had already been implemented in LXGram.

The places of the grammar that were affected by this change are similar to the ones reported in the message-free LinGO Grammar Matrix announcement. The main differences are the following. First, subject-verb inversion does not distinguish between declarative sentences and yes-no questions in Portuguese and realization of the subject does not separate imperatives from other sentences, so we do not need to constrain this feature in subject-head constructions. The second difference is that we are introducing a *qeq* constraint between the topmost LTOP (assuming it denotes the global top) and the main verb's label (LBL), following the literature on MRS ([Copestake et al., 2001]), while the LinGO Grammar Matrix implementation relies on a modification of the scope resolution algorithm in the LKB. Our approach requires a unary syntactic rule to produce the root node of all syntactic trees, for the sole purpose of adding this *qeq* constraint, but the grammar is producing MRSs that conform to the MRS literature.

The type hierarchy used in LXGram for the possible values of the feature SF is very similar to the one in the LinGO Grammar Matrix. We use more verbose names, and there is an extra type that allows underspecification between propositions and commands (for sentences that end with a period). Figure 5.13 shows this hierarchy.

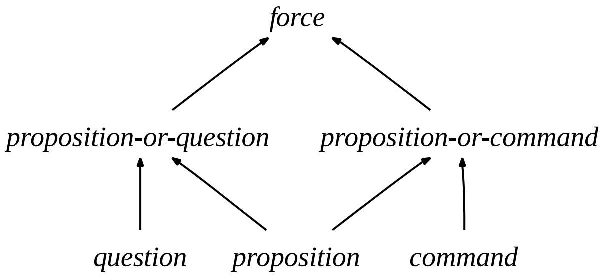


Figure 5.13: Type hierarchy under *force*.

5.5 Variation Between European and Brazilian Portuguese

In LXGram variation between European Portuguese (EP) and Brazilian Portuguese (BP) is taken into account. A mechanism is implemented in order to control grammar behaviour regarding variation between EP and BP.

Currently, only a two-fold distinction is made among the varieties of Portuguese, namely a coarse-grained distinction between EP and BP, although regional differences can obviously be found in the Portuguese spoken in Africa or the rest of the world or in different regions of Portuguese speaking countries. This will be expanded as needed.

The mechanism can be used for several purposes:

1. to restrict the grammar to parse or generate EP sentences only or BP sentences only, or to accept EP and BP but reject sentences showing specific features of both;

2. to use the grammar to detect which variety is used in some text (by parsing it and reading from the resulting feature structures).

It is important that the grammar can work with both EP and BP because of coverage, but accepting the two will necessary increase ambiguity. The ability to control variation is important in that it is a way to control the ambiguity generated from accepting both varieties. It is also very desirable when generating.

The implementation is very simple and is inspired on the idiom detection mechanism implemented in the Jacy grammar and described at http://wiki.delph-in.net/moin/JacyIdiom (see also Section 5.8). A brief introduction to it is presented below. For a detailed account, see [Branco and Costa, 2007].

We make use of a feature VARIANT, which encodes the variety of Portuguese being used. It is appropriate for all signs and declared to be of the type *variant*. Its possible values are presented in Figure 5.14.

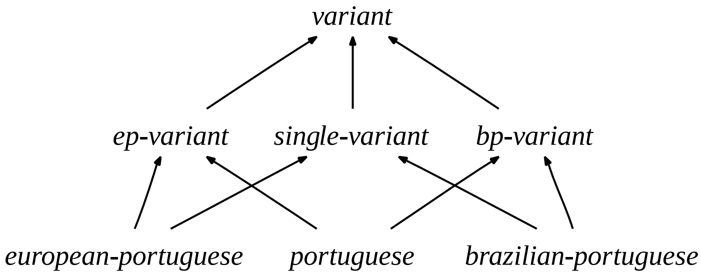


Figure 5.14: Type hierarchy under *variant*.

This attribute is constrained to take the appropriate value in lexical items and constructions specific to one of the two main Portuguese varieties. For example, the lexical entry for the noun *autocarro* (*bus*, exclusive to EP) includes the constraint that the attribute VARIANT has the value *ep-variant* and the corresponding BP entry for *ônibus* constrains the same feature to bear the value *bp-variant*. The only two types that are used to mark signs are *ep-variant* and *bp-variant*. The remaining types presented in Figure 5.14 are used to perform computations or to constrain grammar behaviour, as explained below.

The feature VARIANT is structure-shared among all signs that comprise a full parse tree. This is achieved by having all lexical rules unify their VARIANT feature with the VARIANT feature of their daughter, and all syntax rules identify their VARIANT feature with the VARIANT feature of every daughter.

If two signs (lexical items, syntax rules) in the same parse tree have different values for feature VARIANT (one has *ep-variant* and the other *bp-variant*), they will unify to *portuguese*, as can be seen in Figure 5.14. This type means that lexical items or constructions specific to two different varieties are used together. Furthermore, since this feature is shared among all signs, it will be visible everywhere, for instance in the root node.

It is possible to constrain the feature VARIANT in the root condition of the grammar (i.e. in the definition for the type that corresponds to the LKB's ***start-symbol*** parameter and specifies the constraints that the root node of a valid parse must satisfy). If this feature is constrained to be of the type *single-variant*, the grammar will accept either EP or BP, but the sentences with properties of both will be blocked. As explained in the previous paragraph, the feature VARIANT will have the value *portuguese* in this case, a value incompatible with *single-variant*. If this feature is constrained to be of the type *european-portuguese*, the grammar will not accept any sentence with features of BP, since they will be marked to have a VARIANT of the type *bp-variant*, which is incompatible with *european-portuguese*. It is also possible to have the grammar reject EP (using type *brazilian-portuguese*) or to ignore variation completely by not constraining this feature in the start symbol.

It is not practical to have to change the grammar code in order to be able to switch among variation policies, and that would also require reloading the grammar for changes to take effect. For this reason, the most convenient way of changing the variation control policy followed by the grammar is to change the LKB parameter ***start-symbol***. To make it simple to change, we provide four root types that correspond to the four types of behavior we anticipate. They are all defined in the file `roots.tdl` and are presented in Figure 5.15. They have the common supertype *root-clause*, which encapsulates the remaining constraints on valid root nodes.

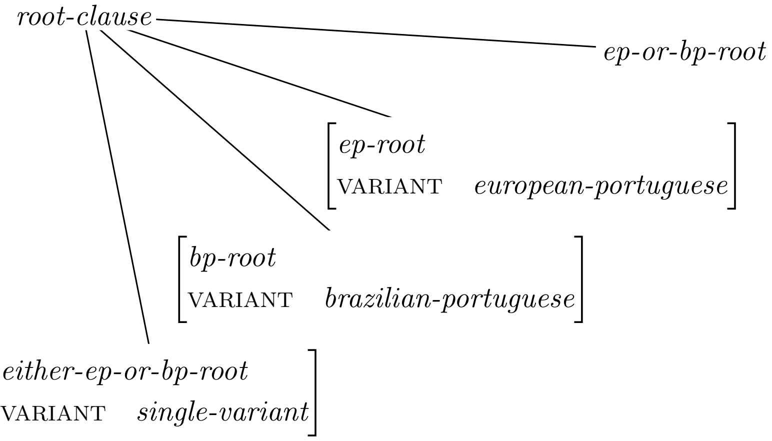


Figure 5.15: Types defining variation control policies.

Choosing *ep-and-bp-root* as root condition has the effect that no constraint is put on variants. Choosing the type *only-ep-root* makes the grammar reject all sentences with features of BP; *only-bp-root* makes the grammar reject all sentences with features of EP. Type *only-ep-or-only-bp-root* constrains it to accept EP and BP, but block sentences with properties of both.

In the LKB configuration file `lkb/globals.lsp`, ***start-symbol*** is initialized with the value *only-ep-or-only-bp*, so by default the grammar will accept either EP or BP but reject sentences with variant switching.

It is possible to change the value of the parameter ***start-symbol*** in the Lisp buffer by invoking **(defparameter *start-symbol* '(only-ep-root))**, for instance. Any of the four types just presented will be a valid choice. It is not necessary to reload the grammar for the change to take effect.

It is important to note that the LKB does not unify a type defined as a ***start-symbol*** with the feature structures resulting from parsing or generation. Instead, it only checks if they are unifiable. This means that if the default value of ***start-symbol*** is used (thereby constraining the VARIANT attributes of valid structures to be unifiable with *single-variant*), and a sign with features of EP is present in the input, the resulting feature structure will have VARIANT features of type *ep-variant*, not *european-portuguese*, although *european-portuguese* is the most general unifier of *ep-variant* and *single-variant*.

As mentioned in the beginning of this section, it is possible to use the grammar to detect to which variety input text belongs to. This is done by parsing that text using the default start symbol and reading the value of attribute VARIANT from the resulting feature structure: the value *variant* indicates that no variant-specific marked signs were detected and the text could be from both; the values *ep-variant* and *bp-variant*, in turn, result from parsing text with features specific to EP and BP respectively.

5.5.1 Example

With this setup (and using the default variation policy) and the following lexical entries:

- Specific to EP
  - *dezasseis* (*sixteen*)
  - *bebé* (*baby*)
  - *canadiano* (*Canadian*)
- Specific to BP
  - *dezesseis* (*sixteen*)
  - *bebê* (*baby*)
  - *canadense* (*Canadian*)

the following sentences are accepted:

(5) Ontem nasceram dezasseis bebés canadianos.
yesterday were born sixteen.EP babies.EP Canadian.EP
Sixteen Canadian babies were born yesterday.

(6) Ontem nasceram dezesseis bebês canadenses.
yesterday were born sixteen.BP babies.BP Canadian.BP
Sixteen Canadian babies were born yesterday.

Furthermore, (5) is detected to be EP, and (6) to be BP. The sentences in (7) are rejected.

(7) a. Ontem nasceram dezasseis bebês canadenses.
yesterday were born sixteen.EP babies.BP Canadian.BP
Sixteen Canadian babies were born yesterday.

b. Ontem nasceram dezesseis bebés canadianos.
yesterday were born sixteen.BP babies.EP Canadian.EP
Sixteen Canadian babies were born yesterday.

5.6 Pragmatics

LXGram includes some information about pragmatics. Currently, the information encoded there is the information that used to be placed under message relations, identifying:

- topicalized constituents,
- subjects of passives,
- post-verbal subjects.

Pragmatic information is represented under the path SYNSEM|LOCAL|CTXT. The kind of pragmatic information that LXGram includes is encoded in a subfeature of CTXT called BACKGROUND.

Pragmatics is built the same way as semantics, but using these different features. We use the attributes RELS and HCONS under the path SYNSEM|LOCAL|CTXT|BACKGROUND. The information about topicalized constituents, subjects of passives and post-verbal subjects is placed under a relation in SYNSEM|LOCAL|CTXT|BACKGROUND|RELS. This relation is called *theme-rheme_d_rel*, and it has one argument for each of these pieces of information:

$$\begin{bmatrix} \textsc{lbl} & \textit{handle} \\ \textsc{pred} & \textit{theme-rheme\_d\_rel} \\ \textsc{arg0} & \textit{event} \\ \textsc{topicalized} & \textit{non-vacuous} \\ \textsc{postposed} & \textit{handle-or-ref-index} \\ \textsc{passivized} & \textit{handle-or-ref-index} \end{bmatrix}$$

In LXGram, the type *non-vacuous* is the supertype of handles (type *handle*), events (*event*) and referential indices (*ref-ind*). This is the type of the feature TOPICALIZED (for topicalized constituents). The attributes POSTPOSED (for post-verbal subjects) and PASSIVIZED (for subjects of passives) are of the type *handle-or-ref-index*, a supertype of *handle* and *ref-ind*. Each of these features can be left uninstantiated, or filled with an appropriate value.

For instance, in sentences with post-verbal NP subjects, the referential index associated to that NP is also the value of the feature POSTPOSED. In this case, this value is filled in by the Subject-Head construction that projects subjects to the right of the verb. For sentences with pre-verbal or null subjects, this feature is left underspecified. The use of the other features is similar.

The feature PASSIVE holds the index of the subject of passive constructions. The attribute TOPICALIZED is holds the index of topicalized constituents.

This *theme-rheme_d_rel* relation is introduced in the lexical entry for verbs, and there will be one such relation per verb form (at the moment, the exception is with auxiliary verbs, that do not introduce one of these relations, as the past participle that they select for already introduces one). The LBL feature of the *theme-rheme_d_rel* relation is identified with the LBL attribute of the corresponding verb's relation. The event that is the value of the ARG0 attribute of this relation is also identified with the event of the verb.

The composition of pragmatics is parallel to the composition of semantics: the RELS and HCONS of a rule (under BACKGROUND) is the difference-list append of the same features in the daughters.

An important note relates to the filling in of these attributes TOPICALIZED, POSTPOSED and PASSIVIZED. As mentioned above, this *theme-rheme_d_rel* relation is introduced in the lexical entries, but some of these three features are instantiated at a higher syntactic position (in the rule for post-verbal subjects, in the case of the feature POSTPOSED). Therefore, this relation needs to be visible in the feature structures in some place different than the attribute RELS (because in general it is not possible to know the exact position in this list where the relevant *theme-rheme_d_rel* can be found, as there will be one *theme-rheme_d_rel* relation per verb). We put this relation under the path SYNSEM|LOCAL|CTXT|BACKGROUND|KEYS|THEME-RHEME-REL. This feature is percolated from the head daughter to the mother node in all headed syntactic constructions (and it is also percolated from the daughter in all morphological rules): therefore the *theme-rheme_d_rel* relation associated to the verb of the current clause can always be found as the value of this feature in the feature structures for verb-headed constituents.

We present an example for the sentence "chegou o presidente" (*the president has arrived*), with a post-verbal subject, in Figure 5.16.

This information on pragmatics can also be incorporated in the MRSs produced by LXGram.

In LXGram, the root nodes of syntactic trees are produced by a unary syntax rule that takes as daughter a full sentence (the result of this is not shown in the previous example). The main motivation for this rule is related to the composition of semantics: namely it is to introduce in the MRS representation a *qeq* relation between the global top and the handle of the relation corresponding to the main verb.

This syntax rule is implemented with the type *unary-root-clause*, defined in the file `syntax.tdl`. This type *unary-root-clause* is defined as inheriting from *basic-unary-root-clause-no_ctxt_in_mrs*.

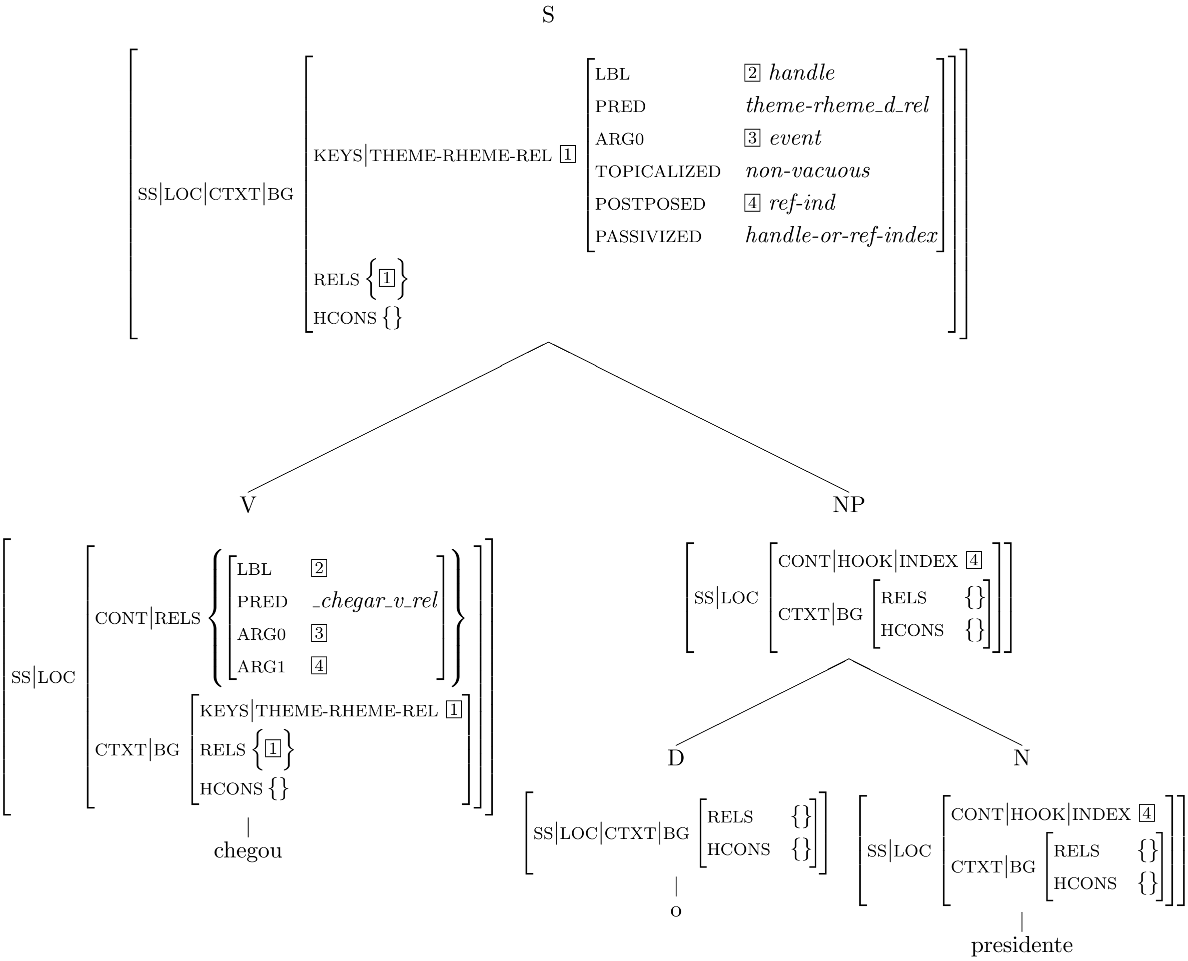


Figure 5.16: Example of the composition of pragmatics. The sentence is "chegou o presidente" (*the president has arrived*). SS abbreviates SYNSEM, LOC abbreviates LOCAL, and BG abbreviates BACKGROUND.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{theme-rheme\_d\_rel}\langle \textit{0:19}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{PASSIVIZED} & \boxed{u6}\ u \\ \text{POSTPOSED} & \boxed{x4}\ x \\ \text{TOPICALIZED} & \boxed{u5}\ u \end{bmatrix}, \begin{bmatrix} \textit{\_chegar\_v\_rel}\langle \textit{0:6}\rangle & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_o\_q\_rel}\langle \textit{7:8}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{RSTR} & \boxed{h8}\ h \\ \text{BODY} & \boxed{h9}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_presidente\_n\_-de-\_rel}\langle \textit{9:19}\rangle & \\ \text{LBL} & \boxed{h10}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y11}\ y \end{bmatrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h3} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h8} \\ \text{LARG} & \boxed{h10} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 5.17: MRS for the sentence "Chegou o presidente" (*The president has arrived*). Information about pragmatics is also included in the MRS.

The constraints on the type *basic-unary-root-clause-no_ctxt_in_mrs* keep the semantic and the pragmatic representations separate. However, there is another type, *basic-unary-root-clause-ctxt_in_mrs*, defined in the same file, that can be used as the supertype of *unary-root-clause* instead. If the supertype of *unary-root-clause* is changed to *basic-unary-root-clause-ctxt_in_mrs*, the information under SYNSEM|LOCAL|CTXT|BACKGROUND will be appended to the MRS representation in SYNSEM|LOCAL|CONT.

In Figure 5.17 we show the MRS that is obtained by including pragmatic information in the semantics. The sentence is the same as that of the previous example: "chegou o presidente" (*the president has arrived*).

This separation of pragmatics and semantics allows the grammar to produce MRS with information that only affects truth conditions. However, if desired, it is also possible to include other sorts of information (namely pragmatic information) that can be useful for applications.

There is however an issue with this implementation concerning generation. When the grammar includes pragmatics in the MRSs, the *theme-rheme_d_rel* relations are introduced in the semantic representations at the topmost syntactic rule based on what is in SYNSEM|LOCAL|CTXT|BACKGROUND. These relations are not in the C-CONT of any rule or in the SYNSEM|LOCAL|CONT of any lexical item. This means that the generator cannot index these relations, and it is not possible to generate from MRSs containing *theme-rheme_d_rel* relations.

5.7 Constructional Content and Constructional Context

As presented in the previous section, the building up of pragmatic information is very much like the composition of semantics.

For the composition of semantics, a feature C-CONT is usually used in order to represent the contribution of lexical and syntactic rules to the semantics. For instance, in syntax rules, the RELS of the mother is the union of the RELS of all daughters and the RELS under the feature C-CONT.

For pragmatics, we could also think of a feature C-CTXT. At the moment, no rule in LXGram contributes to the pragmatic representation, but we would like to keep this possibility open.

However, instead of using a feature C-CONT and another feature C-CTXT, in LXGram we use two features CONT and CTXT grouped under a feature C. The semantic content of a rule is thus represented under C|CONT (instead of C-CONT), and its pragmatic information under C|CTXT.

5.8 Idioms

LXGram implements an idiom detection mechanism similar to what can be found at `http://wiki.delph-in.net/moin/JacyIdiom`. Our implementation is similar to the one described there, and the basic machinery concerning the configuration files that is described there is also used in LXGram. We explain some of the details briefly in the following paragraphs.

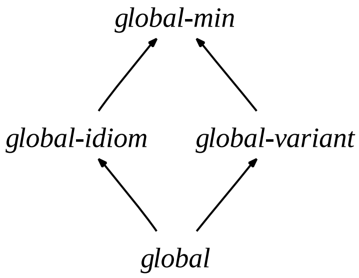
A boolean attribute IDIOM is used. It is present in all feature structures for words and phrases. The value of this feature is unified for all signs comprising a syntax tree. This is accomplished by having all lexical rules identify the value of their attribute IDIOM with that of the same attribute in its daughter and all syntax rules unify their IDIOM feature with that of every daughter.

Because both the features IDIOM and VARIANT (see Section 5.5) are unified across all signs of a syntax tree, we place them under a feature GLOBAL and unify this feature GLOBAL instead, taking advantage of the fact that unification is recursive. We also exploit minimal types (see Section 5.1) in order to avoid the presence of any of these features (IDIOM and VARIANT) if they are left unconstrained.

More specifically, the feature GLOBAL is appropriate for signs and of the type *global-min*:

$$\begin{bmatrix} \textit{sign} & \\ \textsc{global} & \textit{global-min} \end{bmatrix}$$

It can take as values the types in the following hierarchy:



The type *global-idiom* is the type where the attribute IDIOM is declared. It has the following definition:

$$\begin{bmatrix} \textit{global-idiom} & \\ \textsc{idiom} & \textit{bool} \end{bmatrix}$$

In a similar fashion, the type *global-variant* is the most general type for which the feature VARIANT is appropriate:

$$\begin{bmatrix} \textit{global-variant} & \\ \textsc{variant} & \textit{variant} \end{bmatrix}$$

As mentioned before, this feature must be unified across all signs in a parse tree. Therefore, the type *lex-rule* (the supertype of all lexical rules) includes this constraint:

$$\begin{bmatrix} \textit{lex-rule} \\ \textsc{global}\ \boxed{1} \\ \textsc{dtr}|\textsc{global}\ \boxed{1} \end{bmatrix}$$

Also, the types *basic-unary-phrase* (the supertype of all unary constructions) and *basic-binary-phrase* (the supertype of all binary phrases) also include the constraints necessary to propagate these values:

$$\begin{bmatrix} \textit{basic-unary-phrase} \\ \textsc{global}\ \boxed{1} \\ \textsc{args}\ \left\langle \begin{bmatrix}\textsc{global}\ \boxed{1}\end{bmatrix} \right\rangle \end{bmatrix} \begin{bmatrix} \textit{basic-binary-phrase} \\ \textsc{global}\ \boxed{1} \\ \textsc{args}\ \left\langle \begin{bmatrix}\textsc{global}\ \boxed{1}\end{bmatrix}, \begin{bmatrix}\textsc{global}\ \boxed{1}\end{bmatrix} \right\rangle \end{bmatrix}$$

Consider the example idiom in the following sentence:

(8) Ele pregou-lhes um susto.
he nailed them a scare
He scared them.

In order to accommodate such an expression, in LXGram we create a lexical entry for the verb "pregar" according to which it is a ditransitive verb. This lexical entry is also marked with the feature GLOBAL|IDIOM taking the value +.

The machinery described in `http://wiki.delph-in.net/moin/JacyIdiom` makes the LKB parser check for an idiom rule whenever a root node is produced with the value + for the IDIOM feature (as in this example). The parse is accepted only if there is a rule that can match the MRS for the sentence. For this expression, we also include the following rule in the file `idioms.mtr`:

$$\begin{bmatrix} \textsc{input}|\textsc{rels} \left\{ \begin{bmatrix} \textsc{pred} & \text{“\_pregar\_v\_i\_rel”} \\ \textsc{arg2} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \textsc{pred} & \text{“\_susto\_n\_rel”} \\ \textsc{arg0} & \boxed{1} \end{bmatrix} \right\} \end{bmatrix}$$

This rule matches the MRS for this sentence, and the sentence is accepted and marked as containing an idiomatic expression.

A similar sentence with a different noun will not be parsed, because it will display an IDIOM attribute with the value +, but no idiom rule will be found that can match the MRS for that sentence.

Several other expressions can be parsed with this rule and this special entry for the verb "pregar": "pregar dois sustos" (*scare twice*, lit. *inflict two scares*), "pregar um grande susto" (*scare for real*, lit. *inflict a big scare*), etc.

5.9 Punctuation

LXGram contains an incipient implementation of punctuation. At the moment only periods (`.`), question marks (`?`), commas (`,`), ellipses (`...`), exclamation marks (`!`) and quotation marks (`'` and `"`) are recognized, all other punctuation characters being ignored. Except for commas and quotation marks, they are allowed at the end of sentences only. Also, at the moment, commas are constrained only in a few places (for instance, after sentence initial interjections the implementation makes them obligatory — see Section 7.6), and the implementation accepts them in practically all syntactic contexts. Quotations marks are represented in the feature structures, but they are ignored in syntax.

As in the ERG, punctuation is implemented as affixation. This strategy is possibly more involved than treating punctuation as separate tokens, but it produces more familiar parse trees (modulo the extra nodes for the spell-changing rules that attach punctuation characters to base forms), since punctuation does not give rise to independent branches.

The implementation is as follows. A feature PUNCT is used in all signs. There are two subfeatures of PUNCT: LEFT and RIGHT. The attribute RIGHT contains information about the punctuation marks appearing right after a word. The attribute LEFT of a word contains information about the punctuation marks attached to the left of that word (e.g. quotation marks). Under the features LEFT and RIGHT there are three features: INNER-PUNCT, MIDDLE-PUNCT and OUTER-PUNCT. MIDDLE-PUNCT is where quotation marks are represented. INNER-PUNCT is where punctuation appearing between the word and a quotation mark is represented, and OUTER-PUNCT encodes punctuation marks that are more peripheral than quotation marks.

In all lexical entries, the features MIDDLE-PUNCT and OUTER-PUNCT under RIGHT have the value *no-punctuation-mark*, denoting the fact that words come in the lexicon with no punctuation marks attached to them. There is one morphological rule for each punctuation mark. These rules constrain the RIGHT|OUTER-PUNCT feature of the mother node with one of several values: *period* (for the rule inserting a period), *question-mark*, etc. The rules for quotation marks constrain the features RIGHT|MIDDLE-PUNCT and LEFT|MIDDLE-PUNCT in a similar way. No rules have been implemented to constrain the INNER-PUNCT features, but they are left underspecified and can be filled by an external morphological analyzer via e.g. the PET Input Chart input format.

The syntax rules percolate the information in PUNCT. The RIGHT of the mother node of a syntax rule is always the RIGHT of its rightmost daughter. The LEFT of the mother node of a syntax rule is always the LEFT of the leftmost daughter, also. The LEFT|OUTER-PUNCT of the rightmost daughter of binary phrases is the RIGHT|OUTER-PUNCT of the leftmost daughter.

We make sentence final punctuation have an impact on the semantics. In particular we want to distinguish among propositions, imperatives and questions. Note that with the setup described in the previous paragraph, the RIGHT of the root node is the RIGHT of the final word of the sentence. If that word has a period attached to it, the value of this feature will be *period*. We use a subfeature S-FORCE under these types for punctuation, where the associated sentence force is represented. In the case of the type *period*, the value of its feature S-FORCE is *proposition-or-command* (this type is made more specific with information coming from the verb form). This feature is unified in root nodes with the feature SF that appears in MRSs under the event for the main verb (see Section 5.4).

LXGram also accepts sentences with no punctuation mark on the last word. In this case, the sentence is considered as if it ended with a period.

5.10 Agreement

In LXGram agreement is controlled via the unification of the features AGR, where the information relevant for agreement is encoded (person, number and gender). In LXGram the feature AGR is put under SYNSEM|LOCAL|CAT|HEAD.

Agreement is enforced by unifying the AGR features of the relevant elements. For instance, determiners select the noun via their SYNSEM|LOCAL|CAT|MARKER|SELECT feature. The masculine singular form of the definite article, "o", therefore constrains the AGR feature of the noun it selects in the expected way:

$$\left[\text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{HEAD}|\text{MARKER}|\text{SELECT}|\text{LOCAL}|\text{CAT}|\text{HEAD}|\text{AGR}\begin{bmatrix}\text{NUMBER} & \textit{singular}\\ \text{GENDER} & \textit{masculine}\end{bmatrix}\right]$$

However, constraints like these are to be produced by inflectional rules, namely the ones responsible for gender and number inflection. Note that for nouns, these rules constrain the values of the features NUMBER and GENDER under the path SYNSEM|LOCAL|CAT|HEAD|AGR. In LXGram we use the same set of rules to produce gender and number variants of nouns, determiners, adjectives and other elements. To this end, we make nouns specifiers and noun modifiers also contain an AGR feature under their head, which is unified in the lexical entries for these items with the AGR feature of the noun they select. So for instance, determiners have constraints like the following:

$$\begin{bmatrix} \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} \begin{bmatrix} \textsc{agr}\ \boxed{1} \\ \textsc{select}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\ \boxed{1} \end{bmatrix} \end{bmatrix}$$

The inflectional rules simply constrain the person and number features under SYNSEM|LOCAL|CAT|HEAD|AGR. Because this AGR feature of determiners, adjectives, etc. is unified with the AGR feature of the noun they select, the appropriate constraints are put in a place where they will be unified with the AGR of nouns when syntax rules are applied. This way, the same set of morphological rules can apply to a wide range of syntactic categories: nouns, adjectives, determiners, etc.

5.11 Additional Features for the Composition of Semantics

In LXGram we use an additional feature SARG under the path SYNSEM|LOCAL|CONT|HOOK. This feature is either unified with the attribute LTOP or the attribute INDEX under the same path.

The purpose of this feature is to allow for a single lexical entry for verbs that can take an NP or CP complement. Consider the examples with the verb “dizer” below.

(9) a. Ele disse [$_{\text{NP}}$ isso.]
he said that
He said that.

b. Ele disse [$_{\text{CP}}$ que ia chover.]
he said that went rain
He said that it was going to rain.

For the first example, the second argument of the verbal relation is a referential index, found under the INDEX feature of the NP complement. For the second example, this argument is a handle, found in the LTOP attribute of the CP complement.

In LXGram we include a constraint in the lexical type of verbs like this one according to which the second argument of the verb's relation is the SARG feature of its complement. In the lexical entries of nouns, the SARG feature is unified with the attribute INDEX. These features are percolated from the head daughter in syntax rules. In the lexical entries of complementizers, the SARG feature is identified with the feature LTOP.

5.12 Modification

All modification structures are handled via the Head-Functor rules (see Section 5.3).

In LXGram we use a feature under SYNSEM|LOCAL|CAT called MODIFICATION. There are several features under MODIFICATION, each with a particular purpose.

One attribute under MODIFICATION is MODIFIABLE. This feature is used to control spurious attachment ambiguity. Consider the following examples:

(10) a. Chegou ontem um carro.
arrived yesterday a car
A car arrived yesterday.

b. Chegou um carro ontem.
arrived a car yesterday
A car arrived yesterday.

c. Chegou ontem.
arrived yesterday
It arrived yesterday.

In order to parse the first two sentences, in LXGram we allow adverbs to modify verb phrases (as in the first example) and also sentences (second example). Null subjects, like in the last example, are processed with a unary syntactic rule that discharges the SUBJ feature of the daughter and adds pronominal semantics. This creates two possible attachments for the last sentence: the adverb can attach to the verb phrase and the resulting node feed the unary rule for null subjects, or it can attach to the node produced by the unary syntax rule for null subjects. The attribute MODIFIABLE is used to block modification of the mother node of null subject phrases, in order to avoid this potential spurious ambiguity.

Another attribute under MODIFICATION, MODIFIERS, contains information about whether the current syntactic node includes modifiers. This information is used to control the co-occurrence of definite articles and proper names: some proper names cannot be preceded by definite articles unless they are modified (see Section 8.5.2).

Chapter 6

Preprocessing and Morphology

This chapter deals with issues related to LXGram input and treatment of morphology.

6.1 Input

LXGram uses the UTF-8 character encoding. Running it in an environment with a different character encoding will make the use of special characters problematic.

Implementation of morphology is only demonstrative. It is not exhaustive. It should, however, be sufficient to parse and generate many interesting sentences. The plan is to interface LXGram with external components that deal with morphology ([LXS, Web Page; LXL, Web Page; LXC, Web Page; Branco and Silva, 2002; Branco and Henriques, 2003; Branco and Silva, 2003a; Branco and Silva, 2003b; Branco and Silva, 2003c; Branco and Silva, 2003d; Branco and Silva, 2003e; Branco and Silva, 2004a; Branco and Silva, 2004b]), so this aspect will not be further developed.

The file `preprocessor.fsr` includes several preprocessing rules that allow the use of contracted forms in the input. Figure 6.1 shows an example. The formalism for these rules allows the use of regular expressions and references to portions of the matched elements. However, it does not allow the disambiguation of the items involved. For instance, "a" is ambiguous (preposition, article, clitic), but the contraction "à" is not (the expansion "a a" must be a preposition followed by an article), which cannot be expressed in these rules.

6.2 Chart Mapping

LXGram contains an implementation of chart mapping, as described in `http://moin.delph-in.net/Chart_Mapping` and in [Adolphs et al., 2008]. This is the preferred input method when using the grammar with PET. There is a tool (`lxsuite2fsc.jar`) to produce Feature Structure Chart input to PET, which allows the chart mapping machinery to take advantage of information about part of speech, lemmatization and inflectional morphology of word tokens coming from shallow tools (LX-Suite).

The token mapping rules, in the file `chart-mapping/token-mapping-rules-lxsuite.tdl` depend on the output format of LX-Suite and `lxsuite2fsc.jar` (that is, they are sensitive to the specific tag set used). The rest of the chart mapping implementation is insensitive to this: in principle it should work even if the specific shallow tools used change and their ouput is different

```
-ness([ea]s?)([\.\?,]?)    em ess\1\2
```

Figure 6.1: Preprocessor rule to expand the contractions "nesse", "nesses", "nessa", "nessas" into "em esse" (*on that.MASC-SING*), "em esses" (*on those.MASC-PLU*), "em essa" (*on that.FEM-SING*), "em essas" (*on those.FEM-PLU*) respectively.

(different tag set, etc.), provided the major distinctions are still made (coarse part-of-speech categories, inflectional categories appropriate for Portuguese, etc.).

6.3 Verbal Morphology

Tense and aspect information conveyed by verbal inflection may affect truth conditions. Therefore, this information should be reflected in the semantic representations in the grammar. For instance, if the grammar is used for generation, one wants to control these aspects in the grammar input (semantic representations).

Initially, the strategy of coding this information with features and types that reflect morphology directly was adopted. That is, encodings of the semantic values expressed in each occurrence by tense, aspect and modality were not provided in the MRSs produced by the grammar. Instead, the names given to the inflected forms of verbs were used in the semantic representations.

More recently, an analysis of tense and aspect has been implemented in LXGram that makes the semantic representations output by the grammar contain explicit temporal relations between events and times, based on the grammatical tense of verbs as well as the syntactic context where verbs occur.

Details on the analysis and implementation of tense and aspect in LXGram are provided in Chapter 12.

6.4 Nominal Morphology

Morphological gender and number are encoded in the features GENDER and NUMBER respectively. These features show up in the semantic representations of the grammar, a choice that is fairly common in the HPSG literature (e.g. as in [Pollard and Sag, 1994]).

LXGram contains a set of inflectional rules that produce gender and number variants for nouns, adjectives, determiners and pronouns. The same rules apply to items of all of these grammatical categories. They are listed in Chapter 16. There are two sets of rules: one set for gender variants (one rule for masculine forms, one rule for feminine forms, and one rule to produce forms that are underspecified for gender) and another set of rules for number variants (one rule for singular forms, one rule for plural forms and one rule for forms underspecified for number). The gender rules apply directly to lexical items and the number rules apply to the output of the gender rules.

An adjective like “alto” (*tall*) will undergo either the rule to produce masculine forms or the rule to produce feminine forms, and then the rule to produce singular forms or the rule to produce plural forms. The four forms are thus obtained: : “alto” masculine singular, “alta” feminine singular, “altos” masculine plural, “altas” feminine plural.

Items that do not show any gender variants but have number morphology undergo the rule that produces forms that are underspecified for gender and then either the rule for singular forms or the rule for plural forms. The adjective “grande” (*large*) is an example: “grande” masculine or feminine singular, “grandes” masculine or feminine plural. The value of the GENDER feature will be underspecified.

For items that do not have gender or number variants, the rules that produces underspecified values apply. The adjective “simples” (*simple*) falls in this case. This form is underspecified for gender and number.

The application of the inflectional rules is controlled in two different ways, in the files `morphology-for-lkb.tdl` and `morphology-for-pic.tdl`. Only one of these files should be loaded, as they define the same types, but in different ways.

The first file is appropriate when parsing sentences in the LKB. The choice of which rules to use (rules that leave these features underspecified versus rules that constrain the values of these

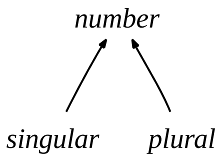


Figure 6.2: Type hierarchy under *number*.

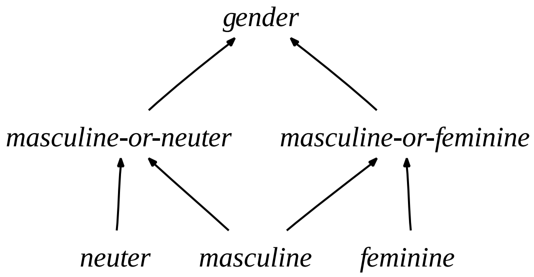


Figure 6.3: Type hierarchy under *gender*.

two morphological features) is controlled lexically (e.g. the lexical entry for "alto", "grande" and "simples", the examples above, have different constraints in this respect, so that the desired rules apply). However, the lexical entries imported from the external lexica do not contain this information, and therefore they will overgenerate (multiple analyses are produced involving underspecified versus specified morphological rules for gender and number) when parsing sentences with the LKB.

It is appropriate to use the second file when parsing sentences with PET using an input format that contains morphological annotations, and also when treebanking sentences that have been parsed with PET in this way. In this case, control over which rules to use depends on the morphological information in that input annotations.

As mentioned, the information about morphological person, number and gender is visible in MRS representations. This information is encoded under the features PERSON, NUMBER and GENDER of referential indices (type *ref-index*). The type hierarchy for the values of PERSON is presented in Section 8.3. It also encodes information about Tu-Vous distinctions. The type hierarchies of gender and number values are presented in Figure 6.2 and Figure 6.3 respectively.

The morphological rules for gender and number unify the AGR feature of the mother with the AGR feature of the daughter.

The hierarchy under *number* comes as expected in as much as Portuguese has a two-fold distinction for number.

The hierarchy under *gender* deserves an explanation. When it comes to adjectives, there is only a two-way distinction for gender values: masculine and feminine. In LXGram, a third value is included, *neuter*, so that the following co-occurrence patterns can be treated as agreement for gender:

(11) a. Beberam todo aquele vinho.
they drank all that wine
They drank all that wine.

b. Beberam toda aquela cerveja.
they drank all that beer

They drank all that beer.

c. Beberam tudo aquilo.
they drank all that
They drank all that.

There are no other combination possibilities between the forms “todo”, “toda”, “tudo” (all meaning *all*) and forms like “aquele”. “aquela”, “aquilo” (all meaning *that*). “Todo” and “aquele” are analyzed as having the masculine gender, “toda” and “aquela” the feminine gender and “tudo” and “aquilo” the neuter gender.

Note that adjectives agreeing with these neuter elements appear in their masculine forms. Also, nouns can never co-occur with “aquilo” or “tudo”.

For these two reasons, we use the type *masculine-or-neuter* in the lexical rules that produce masculine forms (recall that the same rules apply to all classes). We further stipulate that all nouns must have the gender value *masculine-or-feminine*.

As a result, masculine nouns will have the gender value *masculine* (the unifier of *masculine-or-feminine* and *masculine-or-neuter*). Therefore, they cannot co-occur with the neuter forms “aquilo”, “tudo”, etc., which are constrained to have their GENDER feature of the type *neuter*.

On the other hand, the masculine forms of adjectives will have the gender value *masculine-or-neuter*. This way, they can be in an agreement relation with masculine nouns and these neuter elements.

The neuter forms are listed in the lexicon, with the GENDER feature constrained to be *neuter*, and do not undergo the gender inflection rules. Also, the neuter items do not have plural forms. The inflectional rule that produces plural nominal forms constrains the GENDER feature with the value *masculine-or-feminine*. The neuter elements are therefore not elligible to feed the rule for plurals. Furthermore, masculine plural forms of adjectives end up with the value *masculine* for the GENDER feature (the unifier for *masculine-or-neuter*, the type that this feature is constrained to be by the masculine inflectional rule, and *masculine-or-feminine*, the type for GENDER in the plural inflecional rule).

Syntactic agreement is handled by unifying the AGR attribute of the elements involved in an agreement relation.

6.4.1 Evaluatives

LXGram contains an implementation of evaluative forms like the form “animaizinhos” (*poor animals*) in the example below.

(12) Salvem os animaizinhos
save the little/poor animals
Save the little/poor animals

The MRS representation of that sentence is in Figure 6.4.

In LXGram the evaluative forms are produced by a morphological rule. This rule adds the *evaluative_a_rel* relation that is present in the MRS in Figure 6.4.

In the current implementation this rule can apply to nouns, adjectives and past partiples. There is a supertype for the value of the feature HEAD of these elements, used to control the application of this rule.

$$
\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textit{pronoun\_q\_rel}\langle \textit{0:22}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix},
\begin{bmatrix} \textit{pronoun\_n\_rel}\langle \textit{0:22}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix},
\begin{bmatrix} \textit{\_salvar\_v\_-de-\_rel}\langle \textit{0:6}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{x10}\ x \\ \text{ARG3} & \boxed{u11}\ u \end{bmatrix}, \\
\begin{bmatrix} \textit{\_o\_q\_rel}\langle \textit{7:9}\rangle & \\ \text{LBL} & \boxed{h12}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h13}\ h \\ \text{BODY} & \boxed{h14}\ h \end{bmatrix},
\begin{bmatrix} \textit{evaluative\_a\_rel}\langle \textit{10:22}\rangle & \\ \text{LBL} & \boxed{h15}\ h \\ \text{ARG0} & \boxed{e16}\ e \\ \text{ARG1} & \boxed{x10} \end{bmatrix},
\begin{bmatrix} \textit{\_animal\_n\_1\_rel}\langle \textit{10:22}\rangle & \\ \text{LBL} & \boxed{h15} \\ \text{ARG0} & \boxed{x10} \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h13} \\ \text{LARG} & \boxed{h15} \end{bmatrix}
\right\rangle
\end{bmatrix}
$$

Figure 6.4: MRS representation of an evaluative form. The Portuguese sentence is “Salvem os animaizinhos” (*Save the little animals*).

Chapter 7

Basic Phrase Structure

This chapter presents information on the implementation of the basic phrase structure with verbs, adjectives, adverbs, prepositions and conjunctions as heads.

7.1 Constraints on Syntactic Elements

LXGram uses two incompatible subtypes of *sign*, *syntactic-sign* and *morphological-sign* to control the application of lexical rules. Any lexical item that must undergo inflection inherits from the type *morphological-sign* and is thus blocked from feeding the syntactic rules directly. All lexical items that are not inflected and all lexical rules that produce fully inflected items inherit from *syntactic-sign*. These two types are employed in a way similar to the boolean feature INFLECTED that comes in the LinGO Grammar Matrix and the types *uninflected-lexeme* and *fully-inflected-lexeme* where this feature is constrained, but this feature and these two types are not used in LXGram.

The type *synsem-min* is the most general value that the attribute SYNSEM can take, and *optional-synsem-min* corresponds to the LinGO Grammar Matrix type *unexpressed*. The type *optional-synsem-min* is incompatible with the synsem type *canonical-synsem-min* (the equivalent of the LinGO Grammar Matrix type *canonical*), which is the type of SYNSEM that all realized elements are constrained to have (the Principle of Canonicality of [Ginzburg and Sag, 2000]).

The Principle of Canonicality is enforced in LXGram via a constraint on the type of the feature ARGS. The LKB and PET use this feature to encode the daughters of a syntactic or lexical rule. It is list valued, and the order of the elements in this list corresponds to surface word order.

For the implementation of lists, the types *list*, *cons* and *null* come from the LinGO Grammar Matrix. The type *null* represents an empty list, and the type *cons* represents a non-empty list, with the features FIRST (where the first element of that list is) and REST (with the tail of that list, which is also a *list*). The type hierarchy and constraints for lists and lists of optional synsems are similar to the definitions in the LinGO Grammar Matrix for *olist*, but we also use the type *list-of-synsems* for lists whose elements are synsems (the type of the features SUBJ and COMPS in general). The relevant part of the type hierarchy is in Figure 7.1.

The types for non-empty lists of synsems and non-empty lists of optional synsems further constrain these two features in the expected way:

$$\begin{bmatrix} \textit{cons-of-synsems} & \\ \text{FIRST} & \textit{synsem-min} \\ \text{REST} & \textit{list-of-synsems} \end{bmatrix}$$

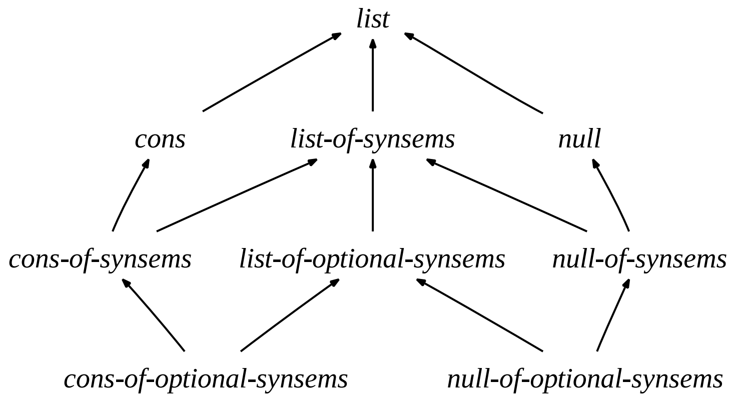
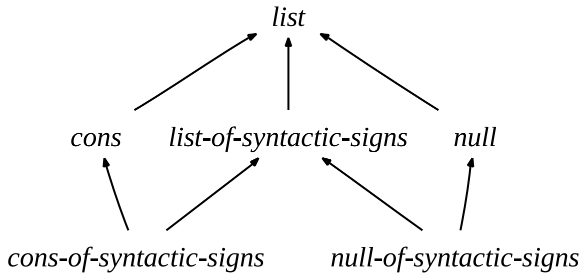


Figure 7.1: Types of lists of synsems

$$\begin{bmatrix} \textit{cons-of-optional-synsems} & \\ \textsc{first} & \textit{optional-synsem-min} \\ \textsc{rest} & \textit{list-of-optional-synsems} \end{bmatrix}$$

In the type *phrase* (the supertype of all syntactic constructions), ARGS is constrained to be of the type *list-of-syntactic-signs*. The relevant part of the type hierarchy is:



The type *cons-of-syntactic-signs* constrains the features FIRST (the head of the list) and REST (its tail), both inherited from *cons*:

$$\begin{bmatrix} \textit{cons-of-syntactic-signs} & \\ \textsc{first} & \textit{syntactic-sign} \\ \textsc{rest} & \textit{list-of-syntactic-signs} \end{bmatrix}$$

The type *syntactic-sign* is constrained to have a synsem of type *canonical-synsem-min*.

As mentioned before, the feature ARGS of the type *phrase* is constrained to be of the type *list-of-syntactic-signs*. In the subtypes where its size is constrained (it will never be empty), the type for ARGS will be inferred to be *cons-of-syntactic-signs*, as this type is the most general unifier of *list-of-signs-syntactic-signs*, a constraint inherited from *phrase*, and *cons*, the most general type for which FIRST and REST are appropriate.

For instance, the constraints defined in the supertype of phrases with two daughters can be very simple:

$$\begin{bmatrix} \textit{basic-binary-phrase} \\ \textsc{args} \left\langle \textit{*top*},\ \textit{*top*} \right\rangle \end{bmatrix}$$

Since *basic-binary-phrase* inherits from *phrase*, where ARGS is declared to be of the type *list-of-syntactic-signs*, the full constraints on *basic-binary-phrase* will be as desired (the definition just presented is notationally equivalent to the left operand):

$$\begin{bmatrix} \textit{basic-binary-phrase} & \\ \text{ARGS} & \begin{bmatrix} \textit{cons} & \\ \text{FIRST} & \textit{*top*} \\ \text{REST} & \begin{bmatrix} \textit{cons} & \\ \text{FIRST} & \textit{*top*} \\ \text{REST} & \textit{null} \end{bmatrix} \end{bmatrix} \end{bmatrix} \sqcap \begin{bmatrix} \textit{phrase} & \\ \text{ARGS} & \textit{list-of-syntactic-signs} \end{bmatrix} =$$

$$\begin{bmatrix} \textit{basic-binary-phrase} & \\ \text{ARGS} & \begin{bmatrix} \textit{cons-of-syntactic-signs} & \\ \text{FIRST} & \begin{bmatrix} \textit{sign} & \\ \text{SYNSEM} & \textit{canonical-synsem-min} \end{bmatrix} \\ \text{REST} & \begin{bmatrix} \textit{cons-of-syntactic-signs} & \\ \text{FIRST} & \begin{bmatrix} \textit{sign} & \\ \text{SYNSEM} & \textit{canonical-synsem-min} \end{bmatrix} \\ \text{REST} & \textit{null-of-syntactic-signs} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The Principle of Canonicality is then simply a constraint on this feature ARGS in a very general type (*phrase*).

7.2 Verb Phrases and Basic Sentence Structure

LXGram provides basic support to VP and sentence structure. VPs are produced from verb items with head-complement schemata that inherit from the type *basic-head-comp-phrase* from the LinGO Grammar Matrix.

Subjects of finite clauses are produced with two schemata that both inherit from *basic-head-subj-phrase* also in the LinGO Grammar Matrix. One is head final, another is head initial.

There are also two rules for null subjects: one for expletives, another for semantically non-empty subjects. In the last case the semantics of a personal pronoun is added to the sentence MRS representation, via the C-CONT attribute of the rule. Figure 7.2 shows an example.

There is also a unary syntactic rule applying to full sentences, which introduces a *qeq* relation between the global top and the handle labeling the relation for the main verb. It produces root nodes.

Subject-verb agreement is controlled in the lexical rules that produce inflected forms of verbs, since these can have access to the subject via the SUBJ valence list.

Subcategorization patterns of verbs are described in Chapter 9.

7.2.1 Postponed Coverage or Known Limitations

Clitics will be properly implemented in phases C.13 and E.19.

7.3 Degree Specifiers

In LXGram almost all adjectives, prepositions and adverbs are allowed to combine with degree specifiers/modifiers. The exceptions are only lexical items that are saliently ungradable, such as some types of scopal adjectives discussed below in Section 7.8.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{pronoun\_q\_rel}\langle \textit{0:6}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle \textit{0:6}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \begin{bmatrix} \textit{\_chegar\_v\_rel}\langle \textit{0:6}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.2: MRS for a sentence with a null subject. The sentence is “Chegou” (*He/she/it arrived*).

The co-occurrence of degree specifiers is however controlled on the basis of gradability categories, which are discussed below, but in general degree specification of all prepositions and adverbs and practically all adjectives is allowed. (13) shows examples of degree specification of prepositions (“a”), adjectives (“interessante”) and adverbs/degree specifiers (“tarde”, “mais”).

(13) a. Estacionei o carro num lugar mais à esquerda.
I parked the car in a place more to the left
I parked the car in a place more to the left.

b. Isso é muito mais interessante.
that is much more interesting
That is much more interesting.

c. Ele chegou muito mais tarde.
he arrived much more late
He arrived much later.

In LXGram intersective semantics is given to degree specification, as can be seen in Figure 7.3, and, as in the ERG, an ARG0 is provided for the relations of the degree specifiers, which allows unambiguous representations of recursive degree specification as depicted in the referred figure. Both “mais” (*more*) and“do que” (*than*) are assigned a single relation (*mais_x_do-que_rel*).

7.4 Conjunction Phrases

At this stage, the “do que” phrase is allowed to introduce nominative NPs only. A more general implementation would also allow full or elliptical sentences, but that is postponed to phase B.9.

Syntactically, the “do que” (*than*) phrase is analyzed as a complement of the degree specifiers “mais” and “menos”, and it must be passed up so that it is projected to the right of the specified head, as the examples in (14) shows.

Furthermore, word order is strict in that the “do que” complement cannot occur between the degree word and its head, so one needs to prevent head-complement rules to apply directly to these items. This cannot be analyzed by constraining the specifier to be a single word, since a degree word is allowed to have degree dependents of its own.

$$\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{matrix}
\begin{bmatrix} \textit{pronoun\_q\_rel}\langle\textit{0:2}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix},
\begin{bmatrix} \textit{pronoun\_n\_rel}\langle\textit{0:2}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix},
\begin{bmatrix} \textit{\_chegar\_v\_rel}\langle\textit{3:10}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix}, \\
\begin{bmatrix} \textit{\_muito\_x\_rel}\langle\textit{11:16}\rangle & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e11}\ e \\ \text{ARG1} & \boxed{e10}\ e \end{bmatrix},
\begin{bmatrix} \textit{\_mais\_x\_-que-\_rel}\langle\textit{17:21}\rangle & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e10} \\ \text{ARG1} & \boxed{e12}\ e \\ \text{ARG2} & \boxed{x13}\ x \end{bmatrix},
\begin{bmatrix} \textit{\_tarde\_a\_rel}\langle\textit{22:27}\rangle & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e12} \\ \text{ARG1} & \boxed{e2} \end{bmatrix}, \\
\begin{bmatrix} \textit{pronoun\_q\_rel}\langle\textit{35:37}\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{RSTR} & \boxed{h15}\ h \\ \text{BODY} & \boxed{h16}\ h \end{bmatrix}
\begin{bmatrix} \textit{pronoun\_n\_rel}\langle\textit{35:37}\rangle & \\ \text{LBL} & \boxed{h17}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{ARG1} & \boxed{y18}\ y \end{bmatrix}
\end{matrix} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h15} \\ \text{LARG} & \boxed{h17} \end{bmatrix}
\right\rangle
\end{bmatrix}$$

Figure 7.3: MRS for the sentence “Eu cheguei muito mais tarde do que tu” (*I arrived much later than you* - lit. *much more late*).

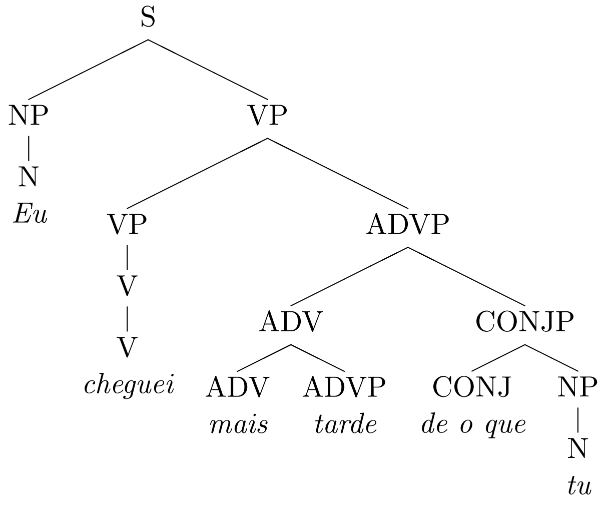


Figure 7.4: Parse tree for sentence (14a).

A third requirement is that, when there are multiple degree specifiers, the appropriate complement must be passed up. So in the example (14c) below it is the "do que" complement of "mais" (*more*) that needs to be copied, not the empty COMPS list of "muito" (*much*).

The example sentences in (14) illustrate these observations, and Figure 7.4 and Figure 7.5 present the parse trees that LXGram assigns to these (where ADVP labels an adverb-headed constituent with empty complements, and ADV an adverb-headed one with non-empty complements).

(14) a. Eu cheguei mais tarde do que tu.
I arrived more late than you
I arrived later than you.

b. * Eu cheguei mais do que tu tarde.
I arrived more than you late

c. Eu cheguei muito mais tarde do que tu.
I arrived much more late than you.
I arrived much later than you.

As mentioned in Section 5.3, in LXGram the notions of specifier and adjunct have been replaced by functors, so degree specifiers are not treated as specifiers but rather as functors. In any case, what is required is that some instances of functor-head schemata will pass the functor complements up, while others pass those of the head. Additionally, one needs to control which functors can undergo which types of head-functor rules, since not all pre-head specifiers and modifiers can have their complements projected to the right of the head they attach to. This will also solve the third desideratum: "mais" (*more*) must have its complements percolated, "muito" (*much*) must not.

The analysis that is implemented makes available two versions of head-final Head-Functor phrases. One of these versions percolates the complements of the head (*functor-head-hcomps-phrase*), the other version percolates the complements of the functor (*functor-head-fcomps-phrase*). Head-initial Head-Functor constructions always percolate the complements of the head. The relevant nodes in the tree in Figure 7.5 are decorated with the names of the phrases involved. Rule application is controlled via an additional feature: COMPS-POSITION. The value of this feature

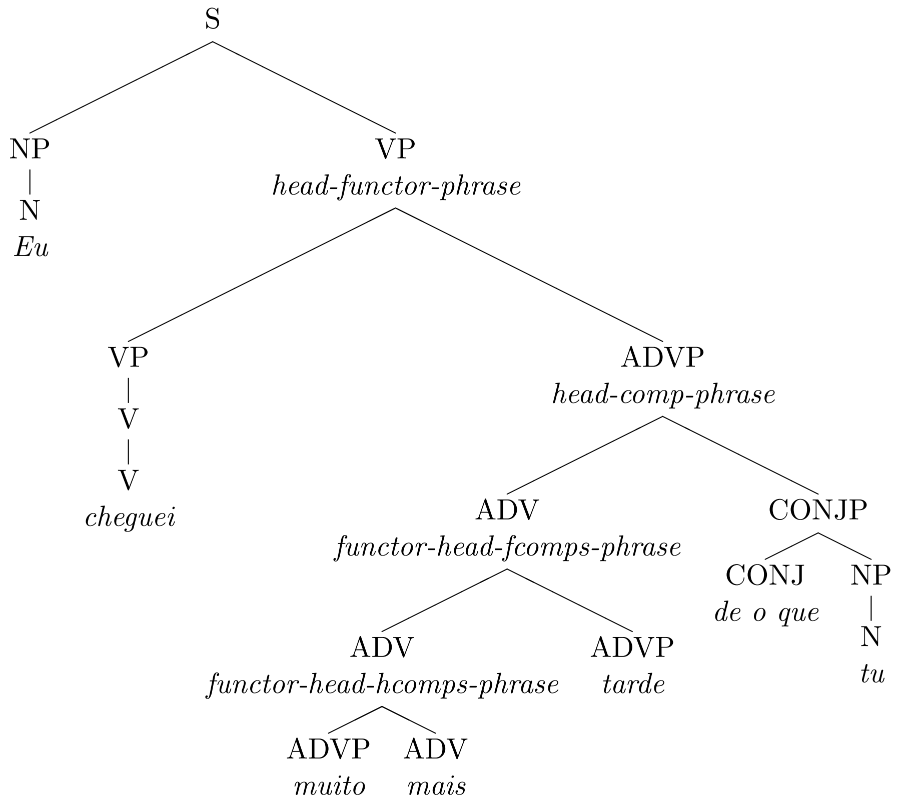


Figure 7.5: Parse tree for sentence (14c).

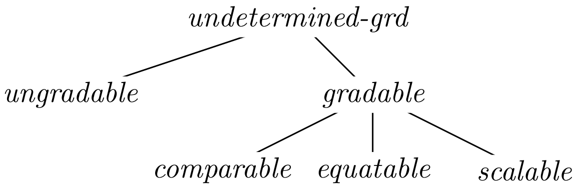


Figure 7.6: Simplified type hierarchy under *undetermined-grd*.

in the functor daughter determines whether *functor-head-hcomps-phrase* or *functor-head-fcomps-phrase* is used.

The sentence in (15) is also considered ungrammatical (it would require some functor phrase that appends the COMPS of both of its daughters).

(15) * O Pedro é mais mais alto do que a Maria do que a Ana.
the Pedro is more more tall than the Maria than the Ana
intended: *Pedro is taller than Maria more than Ana is.*

As was mentioned, constrained recursion of degree specification is allowed. On the one hand, degree specifiers can have a degree specifier of their own. On the other hand, their co-occurrence is constrained. An attribute GRADABLE [1] is employed with a type *undetermined-grd* with the values presented in Figure 7.6.

For instance, to block the co-occurrence of two "mais" (*more*), as in example (15), "mais" selects a head with GRADABLE *comparable* but its own GRADABLE feature is of type *scalable*

[1] GRADABLE is a feature of HEAD.

(specifiable by e.g. “muito” - *much*). An element that can be modified by “tão ... como/quanto” (*as ... as*) has a GRADABLE feature with a value unifiable with *equatable*.

The hierarchy of *undetermined-grd* presented is actually simpler than the one implemented, because some details have been omitted. For example, “mais” must actually constrain the head it selects to be *comparable by “mais”* (i.e. there are subtypes of *comparable*), because some adjectives that have synthetic comparatives cannot form analytical ones (“*mais bom”), but can nevertheless be specified by “menos” (*less*), which also selects a head with a GRADABLE feature of (a subtype of) type *comparable*.

Forms of adjectives that are referred to in traditional Portuguese grammar as absolute superlatives (forms ending in “-íssimo”: “contentíssimo” — *very happy*) are treated in a similar manner. They are of course generated by a lexical rule. It adds a semantic relation similar to that of “muito” (*very*), but it is different from degree specification by “muito” in two mais respects: LXGram does not commit to saying that “-íssimo” suffixation and modification by “muito” are semantically equivalent — the relation introduced by this rule has the name *íssimo_x_rel* (Figure 7.7 shows the MRS for sentence (17a) below) —; its output is constrained to have a GRADABLE feature with the value *ungradable*, whereas “muito” has the value *scalable* (see (16–17)).

(16) a. Estou muito contente.
I am very happy
I am very happy.

b. Estou muito muito contente.
I am very very happy
I am very very happy.

c. * Estou mais muito contente.
I am more very happy

d. * Estou tão muito contente.
I am as very happy

(17) a. Estou contentíssimo.
I am very happy
I am very happy.

b. * Estou muito contentíssimo.
I am very very happy

c. * Estou mais contentíssimo.
I am more very happy

d. * Estou tão contentíssimo.
I am as very happy

7.4.1 Postponed Coverage or Known Limitations

The analysis of degree specifiers began before the functor schemata were implemented, and the first implementation treated them as specifiers, so they are not allowed to modify verbs. With the functor analysis, they are no longer specifiers and modification of verbs by these items can now be implemented without requiring multiple lexical entries. This has not been completed yet.

The current implementation of the “-íssimo” lexical rule applies only to adjectives, but some adverbs can also present these forms (e.g. “cedíssimo”, *very early*; “cedo” *early* is never an adjective in Portuguese)).

$$\left[\begin{array}{ll}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\left[\begin{array}{ll} \textit{pronoun\_q\_rel}\langle \textit{0:19}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{array}\right],
\left[\begin{array}{ll} \textit{pronoun\_n\_rel}\langle \textit{0:19}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{array}\right], \\
\left[\begin{array}{ll} \textit{íssimo\_x\_rel}\langle \textit{6:19}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e10}\ e \\ \text{ARG1} & \boxed{e2} \end{array}\right],
\left[\begin{array}{ll} \textit{\_contente\_a\_-com-\_rel}\langle \textit{6:19}\rangle & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{y11}\ y \end{array}\right]
\end{array} \right\rangle \\
\text{HCONS} & \left\langle \left[\begin{array}{ll} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{array}\right], \left[\begin{array}{ll} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{array}\right] \right\rangle
\end{array}\right]$$

Figure 7.7: MRS for a sentence with an "-íssimo" form. The sentence is "Estou contentíssimo" (*I am very happy*).

7.5 Adverb Phrases

In LXGram non degree adverbs are distinguished in several dimensions:

- whether they can modify nouns and verbs or only verbs

(18) a. O Pedro [comprou um livro] assim.
the Pedro bought a book so
Pedro [bought a book] this way.

b. O Pedro comprou [um livro assim].
the Pedro bought a book so
Pedro bought [a book like this].

c. O Pedro [comprou um livro] depressa.
the Pedro bought a book quickly
Pedro [bought a book] quickly.

This is implemented via constraints on the feature SYNSEM|LOCAL|CAT|HEAD|MARKER|SELECT. Namely, for adverbs in general the sole element in this list has a head of type *noun-or-verb* (a supertype of nouns and verbs), but adverbs that can modify only verbs constrain it to be of the type *verb*.

- whether they can be a predicate selected by the copular verbs

(19) a. Isso é assim.
that is so
That is like that.

b. * Isso é rapidamente.
that is quickly

Since the copular verbs raise their complement's subject (Section 9.1.1), the implementation distinguishes between adverbs that have an empty SUBJ valence list and those that select a subject.

- whether they have PP complements

(20) a. Ele saiu depois.
he went out afterwards
He left afterwards.

b. Ele saiu depois da festa.
he went out after of the party
He left after the party.

Being a question of valence, the implementation obviously distinguishes between adverbs with an empty COMPS list from those with a non empty one. As the examples in (20) illustrate, the complement is always optional. It is also always a PP, but the preposition may vary. The current implementation distinguishes among adverbs with a PP headed by "com" ("a compasso com" *simultaneously with*), "de" (e.g. "depois de" *after*) and by "de" or "a" ("junto a/de" *next to*).

- whether their semantics is intersective ("depressa" in the example below) or scopal ("possivelmente")

(21) a. Ele possivelmente chegou.
he possibly arrived
He has possibly arrived.

b. Ele chegou depressa.
he arrived quickly
He arrived quickly.

Figure 7.8 and Figure 7.9 show the MRSs produced by LXGram for these two sentences.

Intersective modifiers have an empty HCONS difference list and unify the INDEX feature of the synsem inside their SELECT list with the ARG1 of their elementary predication.

Scopal adverbs have an HCONS with a *qeq* that relates their ARG1 with the LTOP of the element in their SELECT attribute.

- whether they trigger proclisis

(22) a. Ele então comprou-o.
he then bought it
He then bought it.

b. Ele já o comprou.
he already it bought
He has already bought it.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{array}{l} \begin{bmatrix} \textit{pronoun\_q\_rel}\langle \textit{0:3}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle \textit{0:3}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \\ \begin{bmatrix} \textit{\_possível\_a\_rel}\langle \textit{4:17}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e10}\ e \\ \text{ARG1} & \boxed{h11}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_chegar\_v\_rel}\langle \textit{18:24}\rangle & \\ \text{LBL} & \boxed{h12}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \end{array} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h11} \\ \text{LARG} & \boxed{h12} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.8: MRS for a sentence with a scopal adverb. The sentence is “Ele possivelmente chegou” (*He possibly arrived*).

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{array}{l} \begin{bmatrix} \textit{pronoun\_q\_rel}\langle \textit{0:3}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle \textit{0:3}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \\ \begin{bmatrix} \textit{\_chegar\_v\_rel}\langle \textit{4:10}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_depressa\_a\_rel}\langle \textit{11:19}\rangle & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e10}\ e \\ \text{ARG1} & \boxed{e2} \end{bmatrix} \end{array} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.9: MRS for a sentence with an intersective adverb. The sentence is “Ele chegou depressa” (*He arrived quickly*).

Control of clitic placement is still embryonic, but there is a common supertype for elements that trigger proclisis (*proclisis-trigger-verbal-adjunct-item*, defined in the file named `lexical-types.tdl`) and another for those that do not (*non-proclisis-trigger-verbal-adjunct-item*, defined in the same file), whose constraints can be changed in the future when a proper analysis of this phenomenon is implemented (phases C.13 and E.19 of the implementation agenda). But lexical items are already marked according to this property.

- whether they can occur preverbally, postverbally or both

(23) a. Eu mal tinha estacionado o carro.
I "mal" (scopal) had parked the car.
I had hardly/just parked the car.

b. Eu tinha estacionado o carro mal.
I had parked the car "mal" (intersective)
I had parked the car badly.

This sort of information is also lexically specified. For the example above there are two lexical entries for "mal": an intersective adverb that can occur only postverbally and a scopal adverb that can occur only preverbally.

Word order between head and functor is implemented by constraints on the type of the feature MARKER of functors (see Section 5.3.2).

7.6 Discourse markers, interjections

LXGram currently supports special items that tend to occur at the beginning of sentences, like "bom", "ora" (*well*) in the examples (24) below.

LXGram clusters them with interjections and greeting words (some examples of these also in (24).

(24) a. Bom, isso é assim.
well, that is like that
Well, that is like that.

b. Ah, está bem.
oh, it is all right
Oh, it is all right.

c. Olá, estás bem?
Hi, are you OK
Hi, are you OK?

These peripheral adverbs, as well as interjections and greeting words, are given the semantics shown in Figure 7.10 (attribute EXCLAMATION is used instead of GREETING for interjections, and there is no corresponding attribute for the peripheral adverbials, these are distinguished by the relation name).

There is a dedicated syntactic rule (*clause_pre-root* in Section 15) to project them to the left of a root sentence. Because of this, they also receive a dedicated head type (*discourse-element* under the type *adv*) so that only they can be the left daughter of this rule (which constrains the left daughter to have this type of HEAD).

$$\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textit{discourse\_rel}\langle 0{:}15\rangle & \\ \text{LBL} & \boxed{h1} \\ \text{C-ARG} & \boxed{e2} \\ \text{L-HNDL} & \boxed{h5}\ h \\ \text{L-INDEX} & \boxed{e4}\ e \\ \text{R-HNDL} & \boxed{h6}\ h \\ \text{R-INDEX} & \boxed{e3}\ e \end{bmatrix},
\begin{bmatrix} \textit{greeting\_rel}\langle 0{:}4\rangle & \\ \text{LBL} & \boxed{h5} \\ \text{ARG0} & \boxed{e4} \\ \text{GREETING} & \textit{olá} \end{bmatrix}, \\
\begin{bmatrix} \textit{pronoun\_q\_rel}\langle 5{:}15\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x8}\ x \\ \text{RSTR} & \boxed{h9}\ h \\ \text{BODY} & \boxed{h10}\ h \end{bmatrix},
\begin{bmatrix} \textit{pronoun\_n\_rel}\langle 5{:}15\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x8} \\ \text{ARG1} & \boxed{y12}\ y \end{bmatrix},
\begin{bmatrix} \textit{\_bem\_a\_rel}\langle 11{:}15\rangle & \\ \text{LBL} & \boxed{h13}\ h \\ \text{ARG0} & \boxed{e3} \\ \text{ARG1} & \boxed{x8} \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h6} \\ \text{LARG} & \boxed{h13} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h9} \\ \text{LARG} & \boxed{h11} \end{bmatrix} \right\rangle
\end{bmatrix}$$

Figure 7.10: MRS of a sentence including a discourse element. The sentence is “Olá, estás bem?” (*Hi, are you OK?*).

7.6.1 Postponed Coverage or Known Limitations

The current implementation is embryonic: it does not support utterances that consist only of a single element of this type and it does not allow for their occurrence in other positions.

7.7 Prepositional Phrases

The lexical types for prepositions in LXGram vary with respect to the following:

- type of complement they take: NP, VP

(25) a. Vou para [o jardim $_{\text{NP}}$].
I go to the garden
I'm going to the garden.

b. Acabei por [ir lá $_{\text{VP}}$].
I ended up by going there
I ended up going there.

At this stage the implementation distinguishes between prepositions taking an NP complement and prepositions with an NP or VP complement.

The VP prepositions that have been implemented so far all lack semantic content (see below), and they can also all take an NP complement as well. Prepositions without content have a HEAD of type *particle*.

This type introduces a feature COMP-HEAD, which is of type *noun-or-verb*, a supertype of *noun* and *verb* and of no other descendant of *head*. The lexical types that implement semantically void prepositions unify this feature with feature HEAD of their complement, so for the time being that complement can only be an NP or a VP.

The type of complement a semantically void preposition takes is thus visible in its HEAD, so that a head that selects a PP argument can distinguish locally between a PP that is semantically an NP and a PP that is semantically a VP.

Two subtypes of *particle* (*particle-np* and *particle-vp*) are used directly by predicators that select PPs with dummy prepositions, since general constraints are enforced in this subtypes. *particle-np* forces its complement to show oblique case, i.e. its COMP-HEAD (which is shared with the HEAD of the complement of the preposition, as stated above) is constrained to have CASE of type *oblique*, where CASE is a feature of *noun* where case is represented. *particle-vp* is constrained to have a COMP-HEAD with feature VFORM of type *infinitivo* (*infinitive*), where VFORM is an appropriate feature of *verb* where mood information is stored; hence other verb forms are blocked.

- whether they have semantic content and whether they can be modifiers

(26) a. Venho do jardim.
I come from the garden
I come from the garden.

b. Gosto do jardim.
I like of the garden
I like the garden.

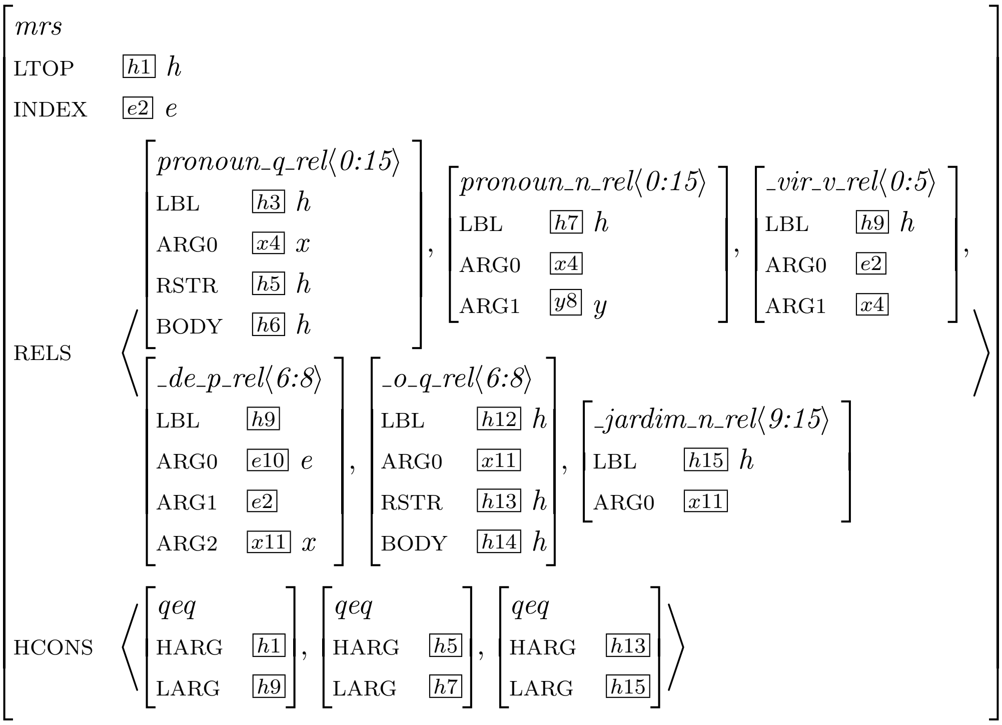

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:15⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:15⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_vir_v_rel⟨0:5⟩ LBL h9 h ARG0 e2 ARG1 x4
_de_p_rel⟨6:8⟩ LBL h9 ARG0 e10 e ARG1 e2 ARG2 x11 x
_o_q_rel⟨6:8⟩ LBL h12 h ARG0 x11 RSTR h13 h BODY h14 h
_jardim_n_rel⟨9:15⟩ LBL h15 h ARG0 x11
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h13 LARG h15

Figure 7.11: MRS for the sentence “Venho do jardim” (*I come from the garden*).

As far as semantic content is concerned, the only implemented distinction is the one between the case when a preposition contributes no semantic relations and the case when it does. For instance, in the example (26b) the preposition “de” is considered empty and in the resulting semantics its complement is an argument of the verbal relation (see Figure 7.12). In the example (26a) the preposition is considered to denote a relation between the situation denoted by the verb and the denotation of “o jardim”, but the exact sense used in particular examples is not determined: “de” is always given a *_de_p_rel* relation. The same applies to all other prepositions. See Figure 7.11 for the full semantic representation given to that sentence by the grammar.

Prepositions that do not contribute any semantic relations are constrained so that they cannot be modifiers, and hence can only appear in a parse tree if they are selected by a head. More specifically, the type *particle* does not inherit from *functor* and hence does not have the feature MARKER.

In order to implement these distinctions, multiple lexical entries are used for prepositions that can sometimes be contentful modifiers and sometimes semantically empty and selected by a head (such as “de”). There are thus two lexical entries for “de”: one that has an empty RELS and a HEAD of type *particle* and a second one with a HEAD of type *preposition* and a non-empty RELS.

Semantically void prepositions also have an extra feature under HEAD, PFORM, where their form is stated. This allows a verb to select for instance a PP whose head is preposition “de” and no other.

The following aspects are also accounted for:

- adjunction sites: nominal vs. verbal
When a PP adjoins to a nominal category, one needs to say that it must follow the head,

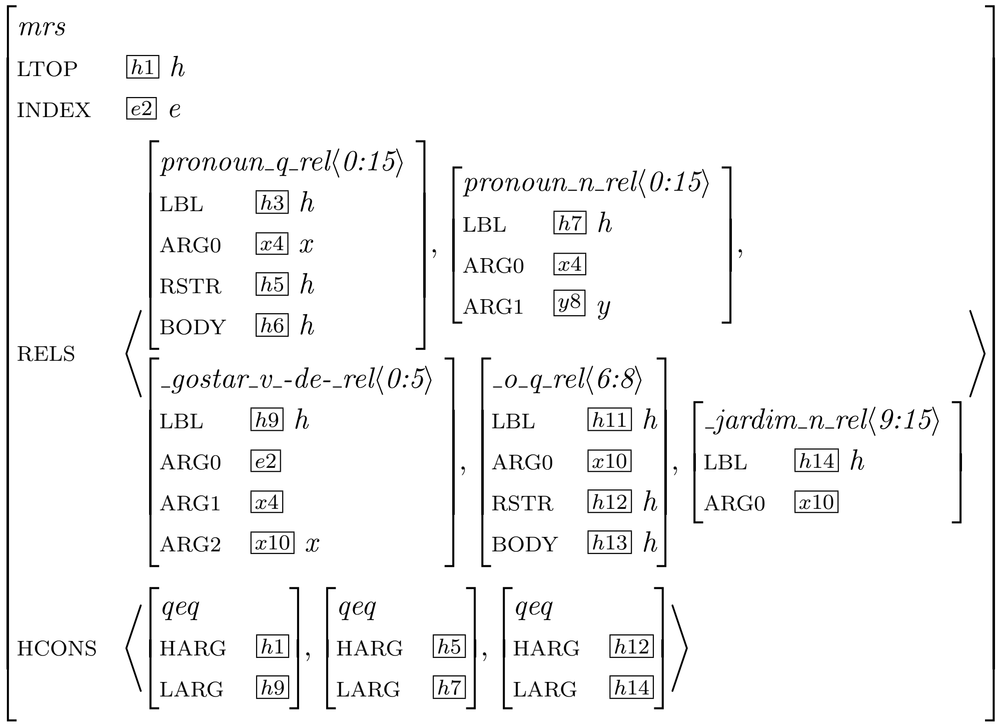

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:15⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:15⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_gostar_v_-de-_rel⟨0:5⟩ LBL h9 h ARG0 e2 ARG1 x4 ARG2 x10 x
_o_q_rel⟨6:8⟩ LBL h11 h ARG0 x10 RSTR h12 h BODY h13 h
_jardim_n_rel⟨9:15⟩ LBL h14 h ARG0 x10
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h12 LARG h14

Figure 7.12: MRS for the sentence "Gosto do jardim" (*I like the garden*).

but when it modifies a verbal projection it can precede it as well. This is implemented through constraints on the features PREHEAD and POSTHEAD explained in Section 5.3.2.

- predicative use
  Semantically vacuous prepositions are prevented from being modifiers and also from being predicates selectable by copular verbs. Semantically vacuous prepositions are treated as raising the subject of their complement, since they can occur with a VP complement and be the complement of raising verbs. So the copular verbs must impose more constraints on their complement besides requiring them to have a subject — see Section 9.1.1. Incompatible HEAD types are currently employed to constrain this. There is an abstract type *predicational-head* that is a supertype of *preposition* (as well as other types that can be the head of the complement of a copula) but not of *particle*.

7.7.1 Postponed Coverage or Known Limitations

Prepositions taking a VP complement and contributing a semantic relation have not been implemented yet.

7.8 Adjective Phrases

Several types of adjectives have been implemented, differing in the following aspects:

- number and type of complements

  The implementation distinguishes between adjectives with zero complements and a one-place relation (ignoring their event argument) and with one complement and a two place relation. There are lexical types for the cases where the complement is a PP headed by "a", "com", "de" and "para" and selecting a noun phrase, illustrated with the sentences in (27).

(27) a. Este computador é igual a esse.
this computer is equal to that one
This computer is equal to that one.

b. O cliente está feliz com a compra.
the client is happy with the purchase
The client is happy with his purchase.

c. Este computador é diferente desse.
this computer is different from that one
This computer is different from that one.

d. Os funcionários são simpáticos para os clientes.
the employees are nice to the clients
The employees are nice to the clients.

The implementation resorts to a PFORM attribute of the preposition. For instance, adjectives that select a complement headed by the preposition "de" taking an NP complement (e.g. "diferente", *different*) are assigned a lexical type where their COMPS list is specified to contain a single element with a HEAD of type *particle-np* (see the Section 7.7) with the feature PFORM of type *de*.

This treatment obviously does not cover all subcategorization frames of adjectives.

- relative word order between adjective and noun

  The relative word order between adjective and noun is controlled by constraints in the lexical entries of adjectives. The implementation is explained in Section 5.3.

(28) a. Tenho um carro amarelo.
I have a car yellow
I have a yellow car.

b. * Tenho um amarelo carro.
I have a yellow car

c. Tenho um fantástico carro.
I have an amazing car
I have an amazing car.

- predicative use

  The possibility of an adjective to be used predicatively or not is also controlled in the lexicon, according to the contrast in the examples below. Since the copular verbs are implemented as raising their complement's subject (Section 9.1.1), adjectives that cannot appear in these contexts are lexically specified to have an empty SUBJ list.

(29) a. O meu carro é amarelo.
the my car is yellow
My car is yellow.

b. * O meu carro é mero.
the my car is mere

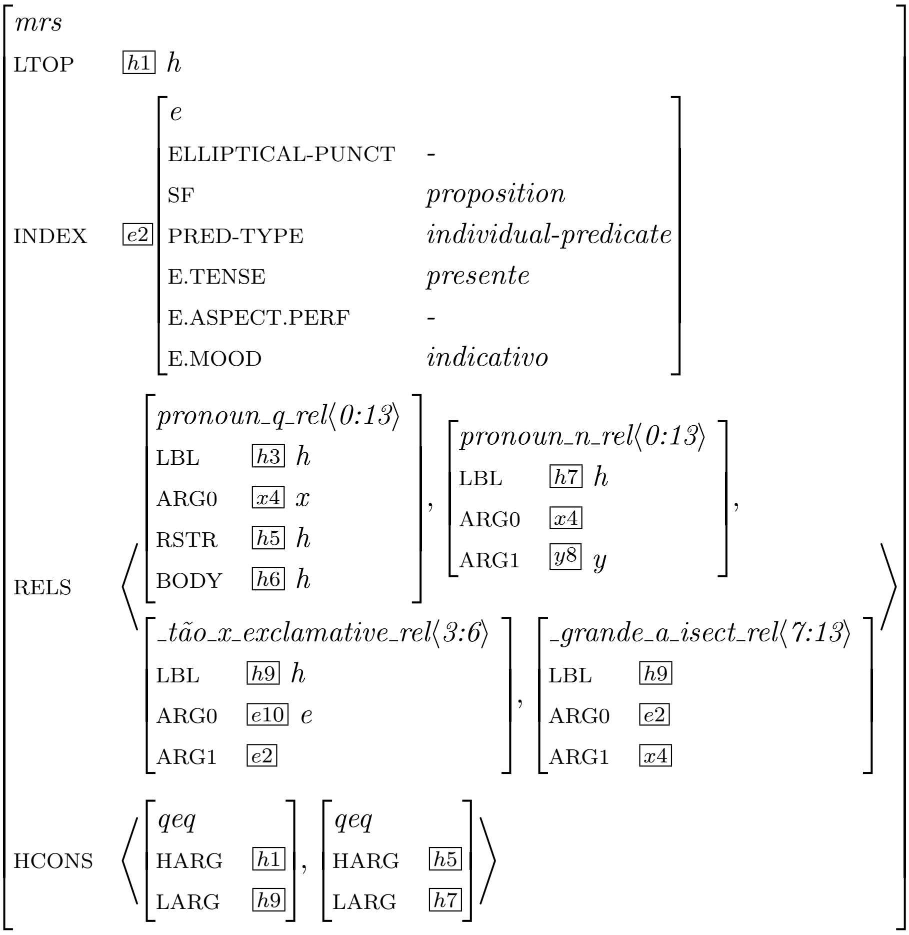


Figure 7.13: MRS for the sentence "És tão grande" (*You are so tall*).

- choice of copular verb

  There are two copular verbs in Portuguese corresponding to English *to be*. One is for individual predicates ("ser"), another for stage-level predicates ("estar"). The sentences in (30) exemplify the difference in meaning.

(30) a. És tão grande!
you are ("ser") so big
You are so tall!

b. Estás tão grande!
you are ("estar") so big
You are so grown up!

Of course, when an adjective or other element that can be selected by a copular verb occurs attributively, it will show an ambiguity between these two readings, so it is important that this information appears in MRSs in a way that allows underspecification.

One such way is to represent it under a dedicated attribute under events. LXGram uses a feature PRED-TYPE to this end. Figure 7.13 and Figure 7.14 show the MRSs assigned to the sentences in (30).

$$\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{\mathit{h1}}\ h \\
\text{INDEX} & \boxed{\mathit{e2}} \begin{bmatrix}
\textit{e} & \\
\text{ELLIPTICAL-PUNCT} & \text{-} \\
\text{SF} & \textit{proposition} \\
\text{PRED-TYPE} & \textit{stage-level-predicate} \\
\text{E.TENSE} & \textit{presente} \\
\text{E.ASPECT.PERF} & \text{-} \\
\text{E.MOOD} & \textit{indicativo}
\end{bmatrix} \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix}
\textit{pronoun\_q\_rel}\langle\textit{0:16}\rangle & \\
\text{LBL} & \boxed{\mathit{h3}}\ h \\
\text{ARG0} & \boxed{\mathit{x4}}\ x \\
\text{RSTR} & \boxed{\mathit{h5}}\ h \\
\text{BODY} & \boxed{\mathit{h6}}\ h
\end{bmatrix},
\begin{bmatrix}
\textit{pronoun\_n\_rel}\langle\textit{0:16}\rangle & \\
\text{LBL} & \boxed{\mathit{h7}}\ h \\
\text{ARG0} & \boxed{\mathit{x4}} \\
\text{ARG1} & \boxed{\mathit{y8}}\ y
\end{bmatrix}, \\
\begin{bmatrix}
\textit{\_tão\_x\_exclamative\_rel}\langle\textit{6:9}\rangle & \\
\text{LBL} & \boxed{\mathit{h9}}\ h \\
\text{ARG0} & \boxed{\mathit{e10}}\ e \\
\text{ARG1} & \boxed{\mathit{e2}}
\end{bmatrix},
\begin{bmatrix}
\textit{\_grande\_a\_isect\_rel}\langle\textit{10:16}\rangle & \\
\text{LBL} & \boxed{\mathit{h9}} \\
\text{ARG0} & \boxed{\mathit{e2}} \\
\text{ARG1} & \boxed{\mathit{x4}}
\end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix}
\textit{qeq} & \\
\text{HARG} & \boxed{\mathit{h1}} \\
\text{LARG} & \boxed{\mathit{h9}}
\end{bmatrix},
\begin{bmatrix}
\textit{qeq} & \\
\text{HARG} & \boxed{\mathit{h5}} \\
\text{LARG} & \boxed{\mathit{h7}}
\end{bmatrix}
\right\rangle
\end{bmatrix}$$

Figure 7.14: MRS for the sentence “Estás tão grande” (*You are so grown up*).

This is implemented by having copular verbs structure-share their event (feature INDEX) with that of their complements (which automatically adds to it the tense, aspect and mood information produced by the inflectional rules the verb undergoes). Additionally, "ser" constrains the PRED-TYPE feature of its event to take the value *individual-predicate*, "estar" constrains it to be *stage-level-predicate*.

The relevant issue here is that adjectives are not constrained for this feature: LXGram lets all adjectives co-occur with both copular verbs.

It is a deliberate decision. Adjectives like "eléctrico" (*electric*) have been reported to not co-occur with "estar", but one can easily find examples where these adjectives do co-occur with "estar", albeit with a shift in meaning ("eléctrico" then means *edgy, restless*). This shift in meaning seems productive.

Of course, a sentence like "esta máquina está eléctrica" (*this machine is electric*, with "estar") is thus strange, but parsed by LXGram. Its strangeness is just a matter of lexical semantics though (it means *this machine is restless*), which is not covered in LXGram.

In any case, the semantic representation given by LXGram captures, however indirectly, the shift in meaning that these adjectives can suffer. Since PRED-TYPE will end up as an attribute of the adjective event (in its ARG0), one can just say that the *electric* sense of "eléctrico" is represented as an *_eléctrico_a_rel* relation with an ARG0|PRED-TYPE of type *individual-predicate* and it *restless* sense is represented by a relation with the same name but an ARG0|PRED-TYPE of type *stage-level-predicate*.

Note also that the *restless* sense of "eléctrico" is availabe with "ser" if the subject is animate. So once again there is an indication that lexical semantics is involved.

- scopal and intersective semantics

  There is also an implemented distinction between intersective ("louro" in examples (31)) and scopal ("competente" in (31)) adjectives.

(31) a. Ele é um pianista louro.
he is a pianist blond
He is a blond pianist.

b. Ele é um pianista competente.
he is a pianist competent
He is a competent pianist.

Figure 7.15 shows the MRS of a sentence with an intersective adjective, and Figure 7.16 the MRS of one containing a scopal adjective.

The examples below (32) show that sometimes this difference is correlated with word order constraints, which is also accounted for in LXGram, by having lexical types for intersective adjectives that can only occur postnominally (the *evil* meaning of "mau") and others for scopal adjectives that can occur prenominally or postnominally (the *bad* meaning of "mau"). There will thus be two lexical entries for an adjective like "mau" in the examples below.

(32) a. Ele é um mau pianista.
he is a bad pianist
He is a bad pianist.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{matrix} \begin{bmatrix} \textit{pronoun\_q\_rel}\langle 0{:}3\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle 0{:}3\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \begin{bmatrix} \textit{identity\_v\_rel}\langle 4{:}5\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{x10}\ x \end{bmatrix}, \\ \begin{bmatrix} \textit{\_um\_q\_rel}\langle 6{:}8\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h12}\ h \\ \text{BODY} & \boxed{h13}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_pianista\_n\_rel}\langle 9{:}17\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{x10} \end{bmatrix}, \begin{bmatrix} \textit{\_louro\_a\_1\_rel}\langle 18{:}23\rangle & \\ \text{LBL} & \boxed{h14} \\ \text{ARG0} & \boxed{e15}\ e \\ \text{ARG1} & \boxed{x10} \end{bmatrix} \end{matrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h12} \\ \text{LARG} & \boxed{h14} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.15: MRS for the sentence "Ele é um pianista louro" (*He is a blond pianist*).

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{matrix} \begin{bmatrix} \textit{pronoun\_q\_rel}\langle 0{:}3\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle 0{:}3\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \begin{bmatrix} \textit{identity\_v\_rel}\langle 4{:}5\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{x10}\ x \end{bmatrix}, \\ \begin{bmatrix} \textit{\_um\_q\_rel}\langle 6{:}8\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h12}\ h \\ \text{BODY} & \boxed{h13}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_pianista\_n\_rel}\langle 9{:}17\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{x10} \end{bmatrix}, \begin{bmatrix} \textit{\_competente\_a\_rel}\langle 18{:}28\rangle & \\ \text{LBL} & \boxed{h15}\ h \\ \text{ARG0} & \boxed{e16}\ e \\ \text{ARG1} & \boxed{h17}\ h \end{bmatrix} \end{matrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h12} \\ \text{LARG} & \boxed{h15} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h17} \\ \text{LARG} & \boxed{h14} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.16: MRS for the sentence "Ele é um pianista competente" (*He is a competent pianist*).

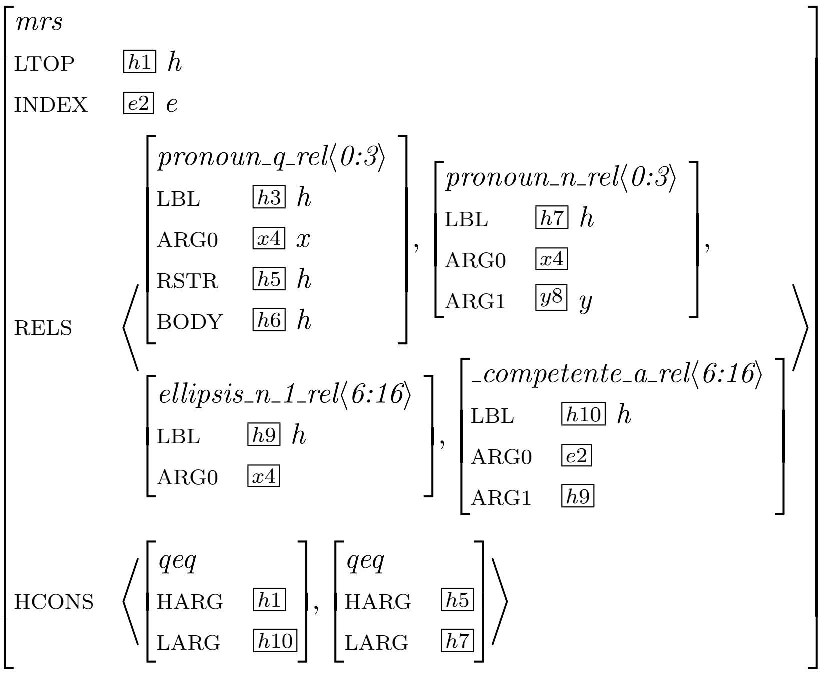


Figure 7.17: MRS for the sentences"Ele é competente" (*He is competent*).

b. Ele é um pianista mau.
he is a pianist bad
He is a bad/evil pianist.

But there are also intersective adjectives that can occur prenominally (as in (33)), as well as scopal ones that occur postnominally (as in (31b)), so there is the need to cross-classify adjectives in these two dimentions, which is implemented by making use of multiple inheritance in the hierarchy of lexical types.

(33) Ele é um famélico pianista.
he is a starving pianist
He is a starving pianist.

For scopal adjectives in predicative contexts, it is necessary to add a relation that can be the argument of these scopal relations, so that the resulting MRSs can be scope resolved. In LXGram this is implemented by forcing these adjectives to feed a unary rule that adds an *ellipsis_n_1_rel* relation, before they combine as the complement of a copular verb. An extra feature MOD-SEM under SYNSEM|LOCAL|CONT|HOOK is used to block scopal adjectives from merging as the complement of copular verbs directly. Figure 7.17 shows the MRS for a sentence with a scopal adjective as predicate.

- synthetic comparatives

There are only a few synthetic comparatives in Portuguese: "maior" — *bigger*, "menor" — *smaller*, "melhor" — *better*, "pior" — *worse*. Therefore, they are simply listed in the lexicon.

They are analyzed as carrying two semantic relations, one adjective relation and another similar to the one of the comparative particle "mais" – *more*.

For example, the semantics of the form "maior" (*bigger*) is said to be *_mais_x_do-que_rel(e0, e1, u2)* ∧ *_grande_a_rel(e1, x3)*, i.e. "more big". Figure 7.18 shows an example MRS.

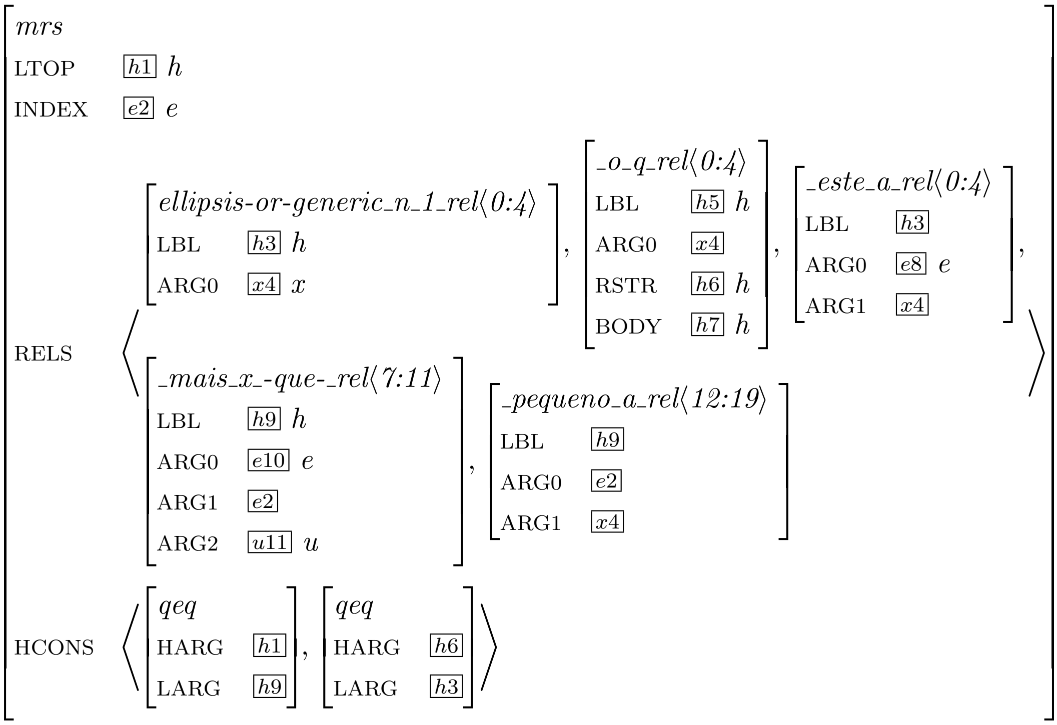


Figure 7.18: MRS for the sentences "Isto é menor" and "Isto é mais pequeno" (*This is smaller*).

The corresponding positive forms are also blocked from producing analytical comparative forms, except for "pequeno" (*small*), for which both comparative forms "menor" and "mais pequeno" are possible. An *ad hoc* feature of HEAD is used for this, since it is in fact exceptional and in no way general (it prevents the occurrence of three adjectives with the item "mais").

Like the comparative "mais", these adjectives also have a "do que" (*than*) complement. The positions where this complement can be projected are described and analyzed in Section 5.3.

- "-íssimo" superlative forms

  For the "-íssimo" superlative forms (e.g. "pequeníssimo" – *very (very) small*, "cansadíssimo" – *very (very) tired*) a lexical rule from lexemes to lexemes is implemented.

  This rule adds a relation similar to the one of the item "muito" (*very*) and produces an ungradable adjective. However, LXGram does not commit to saying that "muito (muito) pequeno" and "pequeníssimo" are exact synonyms - a different name for the added relation is used: *íssimo_x_rel*. See also Section 7.3 and Chapter 16.

- gradability

  A distinction between ungradable adjectives and gradable ones is also present in the lexical types for adjectives.

  As far as this aspect is concerned, however, LXGram is rather permissive, as many adjectives traditionally deemed to be ungradable can be forced to occur with degree specifiers.

  But for illustration purposes, there are types for ungradable adjectives, and the examples below show sentences the grammaticality judgments of which are mirrored in the grammar.

(34) a. Ele é um mero pianista.
he is a mere pianist
He is a mere pianist.

b. * Ele é um muito mero pianista.
he is a very mere pianist

Ungradable adjectives are implemented as having a feature GRADABLE with the value *ungradable* (see Section 7.3).

7.8.1 Postponed Coverage or Known Limitations

A few subcategorization frames of adjectives have been implemented, but LXGram is still missing many possible subcategorization patterns. This work is to be concluded in phases B and C of the implementation.

Chapter 8

Noun Phrases

As far as noun phrases are concerned, LXGram currently implements case distinctions in personal pronouns (described in Section 8.1), some subcategorization frames of nouns (Section 8.2), pragmatic distinctions in second person forms (Section 8.3), non-atomic proper names (Section 8.5), word order constraints between nouns and many of their dependents and NP constituency (Section 8.6) and noun ellipsis (Section 8.7).

8.1 HEAD Types for Nouns

A simplified type hierarchy of *head* values for nouns and pronouns is presented in Figure 8.1.

Common nouns, proper names and personal pronouns are given dedicated HEAD types (*common-noun*, *proper-noun* and *pronoun* respectively). Proper names have a special feature, NAME-SORT, that is not necessary for common nouns and pronouns (see Section 8.5.3 for details).

In the lexical entries of common nouns, proper names and pronouns, only the types *common-noun*, *proper-name* and *pronoun* are used to constrain their HEAD feature.

The head type *noun-or-complementizer* allows a verb like "dizer" (*say*) to select for a complement that is either headed by a complementizer or a noun. The type hierarchy does not make a unifier available for *proper-name* and *noun-or-complementizer*, and therefore proper names are blocked from occurring as the complement of these verbs.

The type *common-or-proper-noun* is a convenient place to put some generalizations: namely common nouns and proper names cannot occur in the contexts where dative pronouns occur, and their CASE feature is constrained to be *non-dative* (see Section 8.1.1).

8.1.1 Case

The feature CASE is a feature of *noun*, a type under *head* that is the value of the feature HEAD of all nominal constituents. It is responsible for constraining the distribution of NPs. The type hierarchy of case values is presented in Figure 8.2.

In Portuguese only personal pronouns overtly manifest case. The leaf types under *case* correspond to the five-fold partition of syntactic contexts induced by these morphological differences. The distinction between *oblique-com* and *oblique-sem* deserves an explanation below.

The remaning types are used to allow underspecification of ambiguous/homonymous forms. Table 8.1 shows the correspondence between case values and some personal pronoun forms.

The type *non-dative* describes the syntactic contexts where NPs headed by a common noun or a proper name can occur: the feature CASE of the head type *common-or-proper-noun* (see the type hierarchy of head types for nouns in Figure 8.1) is constrained to be *non-dative*. Since all nouns (common or proper) have this head type (pronouns have a different one, *pronoun*) no other NP can commute with the dative "lhe" and "lhes" (a PP must be used instead).

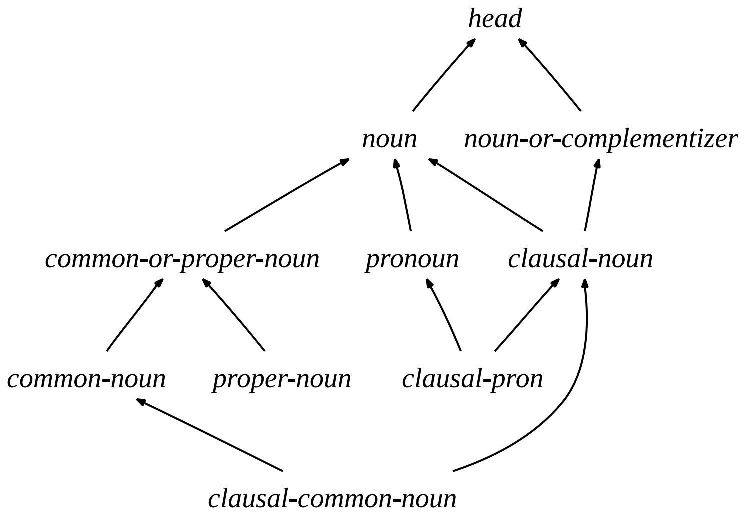


Figure 8.1: Simplified type hierarchy under *noun*.

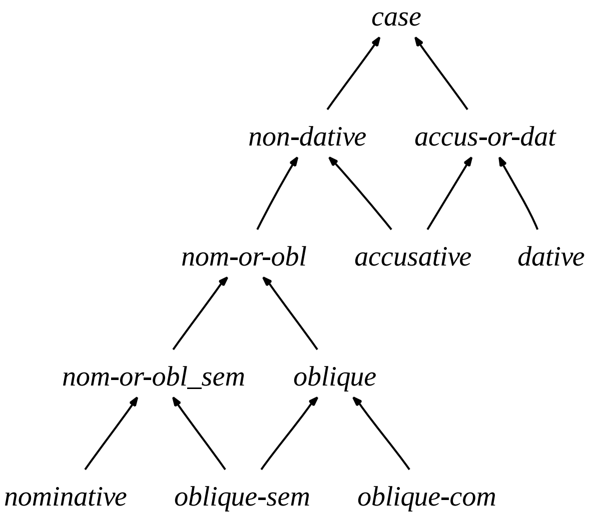


Figure 8.2: Type hierarchy under *case*.

Case	Forms
accusative	o a os as
accus-or-dat	me te vos nos
dative	lhe lhes
nominative	eu tu
nom-or-obl	você vocês
nom-or-obl_sem	nós vós
oblique-sem	mim ti si
oblique-com	migo tigo sigo nosco vosco

Table 8.1: Case values of some personal pronouns.

Four cases are usually recognized for Portuguese: nominative, accusative, dative and oblique. Oblique case is the case of the personal pronoun forms that occur as the complement of prepositions. In LXGram oblique case is divided into *oblique-com* and *oblique-sem*. The former corresponds to the forms “comigo, contigo, consigo, connosco, convosco” (*with me, with you, ...*). The latter case value, *oblique-sem*, is the value for the personal pronoun forms that occur as the complement of other prepositions, among them the preposition “sem” (*without*).

Preprocessor rules are used to split the forms “comigo, contigo, consigo, connosco, convosco” (*with me, with you, ...* into the preposition “com” (*with*) and the forms “migo, ...” (see Table 8.1 for the complete set) and this preposition is constrained to select for NPs with *oblique-com* case. An alternative would be to split these forms into preposition plus the corresponding, standard oblique forms “mim, ti, si, nós, vós”, in which case the distinction between *oblique-com* and *oblique-sem* would no longer be necessary (the usual position). This is however not correct, because the forms “vós” and “vosco” have different T-V values (see Section 8.3), and so a distinction must be made between the two, requiring two separate lexical entries with different constraints on case and T-V values.

With this setup, the expected constraints on case can be implemented: finite verbs constrain the CASE feature of their subject to be *nominative* (in LXGram this is implemented in the inflectional rules responsible for subject-verb agreement), copular verbs taking two NP arguments constrain them both to be nominative, transitive verbs constrain the element in their COMPS list to bear a CASE of type *accusative*, and so on.

8.2 Noun Complementation

As far as noun complementation is concerned, LXGram currently implements nouns with the two subcategorization frames:

- Nouns with a PP complement headed by “de”

  (35) O Pedro é pai da Maria.
  the Pedro is father of the Maria
  Pedro is Maria’s father.

- Nouns with two PP arguments, one headed by “de” and another headed by “de” or “por” The first one will correspond to the complement of a lexically related verb, the second one to its subject, as the example (36) shows.

  (36) a. A destruição da cidade pelo exército foi brutal.
  the destruction of the city by the army was brutal

The army's destruction of the city was brutal.
b. O exército destruiu a cidade brutalmente.
the army destroyed the city brutally
The army brutally destroyed the city.

Other selection frames of nouns will be implemented in the subcategorization phase.

8.3 T-V Distinctions

Tu-Vous (T-V) distinctions, like French *tu* vs. *vous* (when addressing a single person), are represented in the MRSs produced by LXGram. This information does not affect truth conditions and is of a pragmatic nature, but we include it in the semantics.

There are two reasons to make T-V distinctions visible in the semantics: (1) it is important to maintain this sort of information in some applications (e.g. machine translation), and (2) it is very difficult to separate it from the semantics (when the semantics includes person information) if one wants to be able to underspecify certain combinations of person and T-V values. For instance the verb form "é" (*is*) can be a third person singular form or a pragmatically marked second person singular form. If one wants to underspecify this information, so that it does not give rise to two different analyses, person and T-V information must be placed together (i.e. in the same feature) and if person is to be visible in the semantics, so will T-V distinctions.

In LXGram the ability to underspecify forms like "é" was chosen over multiplying analyses, with the disadvantage that some pragmatic information will be visible in the produced MRSs.

The implemented level of T-V differences is course-grained. Here are some distinctions currently ignored:

- First person majestic plurals ("Dizemos", lit. *we say* but meaning *I say*). They were used by royalty in the past, and can also be used by authors.

- Second person plural verb forms and corresponding personal pronouns (like "vós") used as singular. This use is archaic. The grammar considers all these forms plural.

- First person NPs headed by nouns (e.g. someone says "o avô" (lit. *the grandfather*) instead of "eu" (*I*) when addressing his grandchildren). They are mostly confined to nouns describing family relations. They present complications in that they go with third person verb forms, but a third person verb form with a null subject is never interpreted as first person.

The following phenomena are accounted for:

- Three way distinction for second person singular forms: "tu" (akin to French *tu* and German *du*), "você" (intermediate) and NPs headed by a common noun, like "o senhor" (lit. *the Mr.*), akin to French *vous* and German *Sie*.

- Lexical marking of whether a noun can have second person readings (along with third person readings), like "pai" (*father*), "senhor" (*Mr.*), "arquitecto" (*architect*), "Pedro" (a person's name), or is confined to third person interpretations, like "pianista" (*pianist*), "homem" (*man*), "piano" (*piano*), "França" (*France*).

- Unavailability of second person readings in indefinite NPs headed by nouns for which those readings are otherwise available: "o pai" (lit. *the father*) can be second person or third person, but "um pai" (*a father*) is third person.

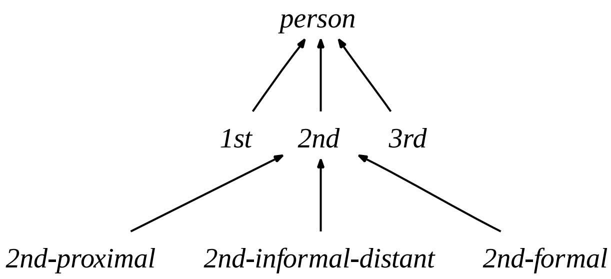


Figure 8.3: Simplified hierarchy for values of person and T-V distinctions.

- Unavailability of second person readings in restrictively modified NPs headed by nouns for which those readings are otherwise available: "o senhor" (lit. *the Mr.*) can be second person or third person, but "o senhor do fato escuro" (*the gentleman in the dark suit*) is third person.

The implementation consists of elaborating the type hierarchy for person values so that it includes both person and T-V categories, and using the same feature PERSON that was previously used only for person to encode T-V distinctions as well. This choice is motivated by the fact that person markings and T-V marking are highly connected in Portuguese: e.g. only second person shows T-V differences at a general level of granularity.

A first approach to encoding a three-way T-V distinction for second person forms would thus posit a type hierarchy for person values like the one in Figure 8.3.

All these types are necessary:

- "Eu" (*I*) or the SUBJ element of verb forms like "sou" (*am*) must be constrained to be PERSON *1st.*

- "Ele" (*he*) or the SUBJ element of verb forms like "chove" (*rains*) must be constrained to be PERSON *3rd.*

- "Tu" (*you*, singular) or the SUBJ element of verb forms like "és" (*are*) must be constrained to be PERSON *2nd-proximal.*

- "Você" (*you*, singular) must be constrained to be PERSON *2nd-informal-distant.*

- The constraints on the subject NP and the verb form in a sentence like "O senhor venha cá" (*You come here*, lit. *The Mr. come here*) must be such that they unify to *2nd-formal* in this and similar sentences.

- An imperative plural form like "vejam" (*see*) does not carry any T-V information, despite being second person. The type *2nd* is justified, and not merely an organizational device.

Merging T-V information with person information means that T-V distinctions are involved in agreement relations, like subject-verb agreement. This is desired, but it requires elaborations on this type hierarchy, since many verb forms can go with subjects displaying various person and T-V combinations.

Complications have to be made to this hierarchy, because there are several degrees of underspecification that must be dealt with. The final type hierarchy for person and T-V values is presented in Figure 8.4.

The data motivating this hierarchy are:

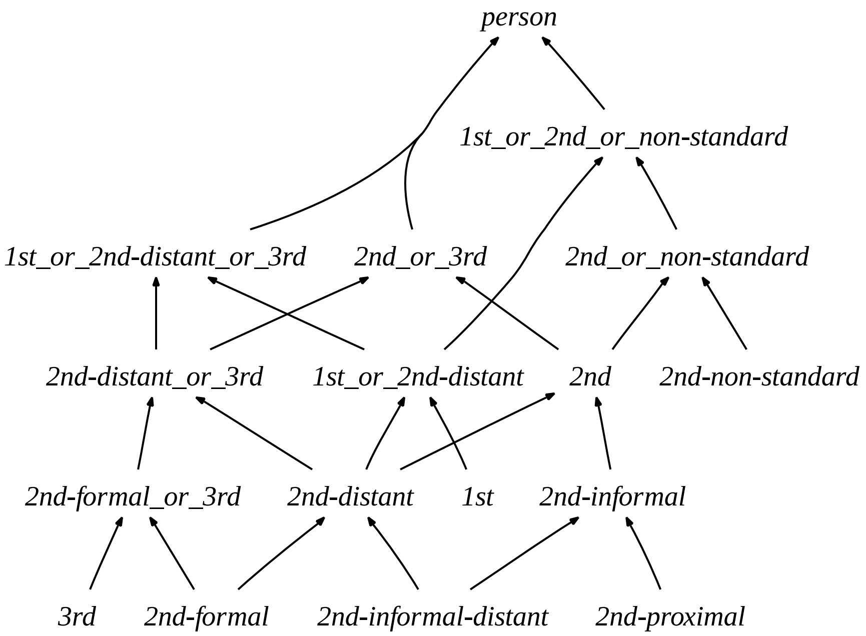


Figure 8.4: Hierarchy for values of person and T-V distinctions.

- The plural item "vocês" corresponds to singular "tu" and "você", so a supertype of *2nd-proximal* and *2nd-informal-distant* is needed: *2nd-informal*.

- An imperative singular form like "veja" (*see*) can be *2nd-formal* or *2nd-informal-distant*, so a supertype *2nd-distant* is needed in order for such verb forms to be underspecified. Non-reflexive "si" is also ambiguous this way.

- The personal pronoun "vós" (*you*, plural) and the verb forms showing subject agreement with "vós" are currently used only dialectally or in contexts where archaic speech is used, like religious ones. Because these verb forms do not take other second person plural subjects (like "vocês" above, or noun headed NPs), an incompatible type must be assigned to them. The type *2nd-non-standard* is used to this end. *2nd-non-standard* and *2nd* are incompatible, as depicted in Figure 8.4, because *2nd* is supposed to mean *real* second person.

- A supertype for *2nd-non-standard* and *2nd* is actually needed, because the forms "vos" (*you*, plural, accusative or dative) and "convosco" (*with you*, plural) correspond to nominative "vocês" (assigned type *2nd-informal*), to plural NPs headed by common nouns with second person interpretations (assigned type *2nd-formal*), and also to nominative "vós" (assigned *2nd-non-standard*). Forms "vos" and "convosco" are hence assigned the value *2nd_or_non-standard*, and considered plural.

- Several noun headed NPs, like "o senhor", are ambiguous between *2nd-formal* readings and *3rd* person readings. A common supertype for these two, *2nd-formal_or_3rd*, is thus required in order to have these NPs underspecified for person.

- Singular verbs forms that take as subjects singular NPs like the ones just mentioned also take "você" as subject. Therefore, they must be even more underspecified. *2nd-distant_or_3rd* is used in these forms and has as direct descendants types *2nd-formal_or_3rd* and *2nd-distant*. These are the verb forms traditionally classified as third person singular.

Personal pronoun	English translation	PERSON value
"eu"	*I*	*1st*
"tu"	*you*	*2nd-proximal*
"você"	*you*	*2nd-informal-distant*
"ele"	*he*	*3rd*
"ela"	*she*	*3rd*
"nós"	*we*	*1st*
"vós"	*you*	*2nd-non-standard*
"vocês"	*you*	*2nd-informal*
"eles"	*they* (masculine)	*3rd*
"elas"	*they* (feminine)	*3rd*
"me"	*me*, *myself*	*1st*
"te"	*you*, *yourself*	*2nd-proximal*
"o"	*you*, *him*	*2nd-distant_or_3rd*
"a"	*you*, *her*	*2nd-distant_or_3rd*
"se"	*yourself*, *yourselves*, *himself*, *herself*, *themselves*	*2nd-distant_or_3rd*
"nos"	*us*, *ourselves*	*1st*
"vos"	*you*, *yourselves*	*2nd_or_non-standard*
"os"	*you*, *them* (masculine)	*2nd_or_3rd*
"as"	*you*, *them* (feminine)	*2nd_or_3rd*
"lhe"	*you*, *him*, *her*	*2nd-distant_or_3rd*
"lhes"	*you*, *them*	*2nd_or_3rd*
"mim"	*me*	*1st*
"ti"	*you*	*2nd-proximal*
"si"	*you*, *yourself*, *yourselves*, *himself*, *herself*, *themselves*	*2nd-distant_or_3rd*
"migo"	*me*	*1st*
"tigo"	*you*	*2nd-proximal*
"sigo"	*you*, *yourself*, *yourselves*, *himself*, *herself*, *themselves*	*2nd-distant_or_3rd*
"nosco"	*us*, *ourselves*	*1st*
"vosco"	*you*, *yourselves*	*2nd_or_non-standard*

Table 8.2: PERSON values of personal pronouns, with T-V information

- Plural verbs forms that take as subjects plural NPs like the ones just mentioned also take "vocês" as subject. Note that whereas "você" is *2nd-informal-distant*, "vocês" is *2nd-informal.* These verb forms are thus even more abstract than their singular counterparts and must be given the type *2nd_or_3rd*, a supertype of *2nd-distant_or_3rd* and *2nd.* These are the verb forms traditionally classified as third person plural.

- In some tenses the traditionally called third person singular forms collide with first person singular forms. Forms of the "Pretérito Imperfeito" are one example ("era", *I, he, she, it was*). The type *1st_or_2nd-distant_or_3rd* is used to constrain their subject.

- Type *1st_or_2nd_or_non-standard* is used grammar internally. Imperative forms are constrained to take a subject with this value for PERSON. This way third person imperatives are blocked.

The lexical entries for personal pronouns have their PERSON feature constrained with the values presented in Table 8.2.

$$\begin{bmatrix} \textit{mrs} & \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & \boxed{h17}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{ARG1} & \boxed{s18}\ s \end{bmatrix}, \begin{bmatrix} \textit{string-equals\_rel} & \\ \text{LBL} & \boxed{h17} \\ \text{ARG0} & \boxed{s18} \\ \text{CARG} & \acute{A}\textit{frica} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 8.5: MRS fragment associated with the proper name “África” (*Africa*).

8.4 Personal Pronouns

Personal pronouns are associated with two relations *pronoun_q_rel* and *pronoun_n_rel*. The representation used is employed in several other grammars. We chose to also use it in LXGram, in order to have representations similar to the other grammars.

The relation *pronoun_n_rel* fills the restrictor of the quantifier relation *pronoun_q_rel*, so personal pronouns receive semantics similar to $\lambda P.pronoun\_q(x, pronoun\_n(x), P(x))$.

8.5 Proper Names

8.5.1 Semantics

In LXGram proper names are given a semantics which is slightly different from the one in the LinGO Grammar Matrix and the other DELPH-IN grammars. Instead of a single binary *named_rel* relation between a referential index and a character string, two relations are used: a binary one between a referential index and a string variable, and a second binary relation holding between the same string variable and a string literal. The second relation is intended to mean string equality.

An example of the MRS fragment corresponding to the name “África” (*Africa*) is presented in Figure 8.5. The equivalent expression

$$\lambda x.named(x, s_1) \wedge string\text{-}equals(s_1, ''\acute{A}frica'')$$

is intended to be synonymous to (albeit more verbose than) the standard

$$\lambda x.named(x, ''\acute{A}frica'')$$

with s_1 to be interpreted as existentially quantified and *string-equals* as the equality relation between strings.

The motivation for this is explained in Section 8.5.3.2.

8.5.2 Information in Lexical Entries for Proper Names

Proper names are cross-classified in the following dimensions:

- Gender
- Number morphology;
- Grammatical person;
- Co-occurrence with determiners.

As with common nouns, gender must be specified lexically. There are names with gender variants, like “Francisco”, a masculine proper name, and “Francisca”, a feminine one, and pairs with masculine “-o” and feminine “-a” spellings are recurrent. For these, it would be desirable to produce both forms from a single lexical entry, for reasons of lexical economy, as is done for common nouns. This is currently not the case. The reason is that the CARG feature in the feature structure representing the semantic relation of these nouns should intuitively be identical to their surface form (different for the two names in the pair). To the best of our knowledge, the LKB does not make it possible to derive the value of CARG from the surface form, so elements of pairs like the one presented must be listed in the lexicon. There are thus only three possibilities for proper names regarding gender: masculine (e.g. “Tejo”, *Tagus*, a river), feminine (e.g. “Lisboa”, *Lisbon*, a city) or underspecified (“Castro”, a surname).

The application of morphological rules for number is also controlled in the lexical entries. Some can only be plural (“Estados Unidos”, *United States*; “Canárias”, *Canary Islands*), some have the same form for both numbers (“Egas”, a men’s name; “Íris”, a women’s name; “Borges”, a surname). These do not undergo the lexical rules for inflection in number.

Proper names are classified in the lexicon as giving rise to third person NPs or NPs ambiguous between third person and second person formal (see Section 8.3). The former are constrained with

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}|\textsc{person} \quad \textit{3rd}\right]$$

and the constraint

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}|\textsc{person} \quad \textit{2nd-distant\_or\_3rd}\right]$$

is inherited by the latter.

As far as the co-occurrence of proper names and determiners is concerned, there are three possibilities: (1) a determiner is obligatory (the definite article by default, but may be another), (2) a determiner is optional, and (3) the definite article is not possible. The sentences in (37) illustrate these three patterns.

(37) a. Foi *em/na Grécia.
it was in/in the Greece
It was in Greece.

b. Foi em/na Itália.
it was in/in the Italy
It was in Spain.

c. Foi em/*na Malta.
it was in/in the Malta
It was in Malta.

The way this different behavior is controlled in LXGram is by resorting to the types for quantifier relations. Quantifier relations are visible in the feature structures for all nouns, under the path SYNSEM|LOCAL|CONT|KEYS|QUANT-REL. A noun dependent that introduces a quantifier relation (e.g. a determiner) in the MRS unifies that relation with this feature in their sister node, e.g.:

$$\left[\textsc{synsem}|\textsc{local} \quad \left[\begin{array}{l} \textsc{cat}|\textsc{head}|\textsc{marker}|\textsc{select}|\textsc{local}|\textsc{cont}|\textsc{keys}|\textsc{quant-rel} \ \boxed{1} \\ \textsc{cont}|\textsc{rels} \left\{\boxed{1}\right\} \end{array}\right]\right]$$

The feature KEYS is unified between the mother and the head daughter in all headed constructions, via a constraint added to the type *headed-phrase*, from which all headed constructions inherit:

$$\begin{bmatrix}\text{SYNSEM}|\text{LOCAL}|\text{CONT}|\text{KEYS } \boxed{1}\\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOCAL}|\text{CONT}|\text{KEYS } \boxed{1}\end{bmatrix}$$

This way, the quantifier relation of an NP is always visible in the feature structure for the noun that is the head of that NP.

We use a type hierarchy of quantifier relations to control the co-occurrence pattern of proper names with determiners. An example follows. The construction for bare NPs introduces a quantifier relation of the type *bare-qrel*. A proper name that must be preceded by a determiner, like "Grécia" (*Greece*) in (37a), comes in the lexicon with the its QUANT-REL constrained to be of type *expressed-or-plural-qrel.*[1]

We organize these types for quantifier relations in a hierarchy in such a way that the unifier of *bare-qrel* and *expressed-or-plural-qrel* is constrained to correspond to a plural noun (via a constraint under ARG0|NUMBER inside the quantifier relation).

The other co-occurrence restrictions are controlled in a similar way: proper names that cannot occur with definite articles constraint their QUANT-REL feature with a type that is incompatible with the type of the quantifier relation introduced by definite articles.

Proper names that cannot co-occur with definite articles (when there is no other material in the NP) must nevertheless be preceded by a definite article (or other determiner) if they are modified. Compare the following two sentences:

(38) a. Isso aconteceu em/*na Lisboa.
that happened in/in the Lisbon
That happened in Lisbon.

b. Isso aconteceu *em/na Lisboa de 1700.
that happened in/in the Lisbon of 1700
That happened in 18th century Lisbon.

LXGram employs an attribute MODIFICATION|MODIFIERS under SYNSEM|LOCAL|CAT to control this behavior (see Section 5.12). The two main values that this feature takes are *has-modifiers* and *no-modifiers*. Nouns come in the lexicon with the latter value, and Head-Functor constructions produce nodes with the value *has-modifiers* for this feature.

These types also have a QUANT-REL feature, where a quantifier relation is represented. This relation is unified with the quantifier relation of the NP by determiners and bare-NP constructions. The type of the quantifier relation under *no-modifiers* is such that when unified with the quantifier relation for these names, it results in a type that is incompatible with the type of

[1] A name like this one can in fact appear without a determiner if it is plural. Consider the examples with names of letters:

(1) a. * (O) "A" é a primeira letra do alfabeto.
the A is the first letter of the alphabet
A is the first letter of the alphabet.

b. Essa palavra não tem "A"s.
that word not has As
That word has no A in it.

quantifier relation of definite articles. The type of the quantifier relation under *has-modifiers* is such that the result of unifying it with the type of quantifier relation associated to these names results in a type that is incompatible with the type of quantifier relation introduced by the bare NP constructions.

8.5.3 Name-Name Phrases

By name-name phrases we mean expressions that are the simple concatenation of proper names and have the distribution of a proper name. Some examples are "Fernando Pessoa", "João Sebastião Ribeiro".

8.5.3.1 Syntax

These constructions can be analysed as head-initial. This is compatible with the observation that features like gender percolate from the leftmost daughter, as examples (39) illustrate (in LXGram, the feature AGR, where agreement is constrained, is a feature of HEAD – see Section 5.10).

(39) a. o João
the.MASCULINE João
João (a men's name)

b. a Maria
the.FEMININE Maria
Maria (a woman's name)

c. o João Maria
the.MASCULINE João Maria
João Maria (a men's name)

d. a Maria João
the.FEMININE Maria João
Maria João (a women's name)

The implementation makes use of a dedicated syntax rule for this construction, the *name-name* phrase. It is a binary rule that constrains both daughters to have HEAD features of type *proper-noun* and is head-initial, i.e. the HEAD feature of the mother node is unified with the HEAD of the left daughter.

The number of names in this construction has no upper bound, so the rule must be able to iterate. As stated, it allows both left recursion and right recursion. This produces multiple analyses when more than two names are present, e.g. "António Oliveira Salazar" yields [[António Oliveira] Salazar] and [António [Oliveira Salazar]]. For reasons related to semantic composition (see Section 8.5.3.2), LXGram chooses to allow only right recursion by constraining the left daughter to have a SYNSEM of type *lex-synsem* (most phrases in LXGram, including the *name-name* phrase, produce nodes with a SYNSEM of type *phrase-synsem*, which is incompatible with *lex-synsem*).

The interaction with postnominal modification of names also multiplies parses. Consider example (40).

(40) Aquela não era a Maria João que conhecíamos.
that not was the Maria João that we knew
That wasn't the Maria João that we knew.

No constraints have yet been mentioned to prevent both analyses [[Maria João] [que conhecíamos]] and [Maria [João [que conhecíamos]]]. In LXGram the second analysis is rejected

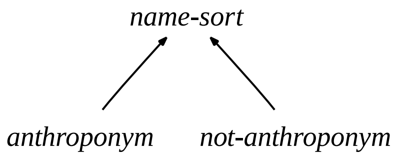


Figure 8.6: Hierarchy for values of feature NAME-SORT.

by a constraint in *name-name* phrases whereby the daughters cannot contain modifiers. More specifically, the left daugher is already constrained to have a *lex-synsem*, so prenominal modifiers are not problematic; the constraint

$$\left[\text{NON-HEAD-DTR}|\text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{MODIFICATION}|\text{MODIFIERS} \quad \textit{no-modifiers}\right]$$

is included to block modifiers in the right (non-head) daughter (the feature MODIFICATION is explained in Section 5.12).

The nouns that can appear in this construction are person names. Other proper names (e.g. names of countries) do not participate in it. In order to control for this, it is necessary to state in the lexical entries for proper nouns whether they denote people names or not. It turns out that this information is already there, under grammatical person (see Section 8.5.2): if a proper name is marked as allowing for second person readings, it is a person's name; otherwise it is not.

The advantage of blocking non-person names from this construction is to prevent multiple analyses for non-ambiguous sentences. Example (8.5.3.1) shows a sentence that would give rise to an illegitimate parse if non-person names were allowed in this construction (with "Lisboa Kiev" as a single name and a null subject).

(41) Como Lisboa Kiev também tem sete colinas.
like Lisbon Kiev also has seven hills
Like Lisbon, Kiev also has seven hills.

It is worth noting that the fact that many place names can also be surnames does not undermine this restriction, because multiple lexical entries are needed in these cases. Multiple lexical entries are necessary, because surnames tend to optionally go with definite articles (see Section 8.5.2) and be underspecified for gender, whereas place names generally have a specific gender value and are more idiosyncratic when it comes to co-occurrence with determiners.

This restriction cannot however be enforced by constraining person values. Since the names that are legitimate here are the ones that are constrained with a PERSON feature with value *2nd-distant_or_3rd* and the ones that are not allowed come in the lexicon with a PERSON that is *3rd*, the only way to control rule application based on this feature would be to say that the daughters of this rule have PERSON constrained to be *2nd*, which is a subtype of *2nd-distant_or_3rd* and has no unifier with *3rd* (see Figure 8.4). But this would constrain the PERSON value of the resulting phrase inappropriately to be *2nd* person (since it is percolated from the head daughter), as the resulting phrase is also underspecified as *2nd-distant_or_3rd.*

Therefore, another mechanism is used to restrict application of this syntactic rule to people's names. A feature NAME-SORT is involved, which is a feature of HEAD appropriate for the type *proper-noun*, a subtype of *head* (see Figure 8.1). NAME-SORT is of type *name-sort*, with the subtypes displayed in Figure 8.6.

Since the PERSON value of proper names must be stated in the lexicon, two supertypes of lexical leaf types can be used to constrain it, *noun-proper-second-or-third-person-item* and *noun-*

$$\begin{bmatrix} \textit{name-name-phrase} & \\ \textsc{synsem} & \begin{bmatrix} \textit{phrase-synsem} & \\ \textsc{local}|\textsc{cat}|\textsc{head}\ \boxed{1} & \begin{bmatrix} \textit{proper-noun} & \\ \textsc{name-sort} & \textit{anthroponym} \end{bmatrix} \end{bmatrix} \\ \textsc{head-dtr} & \boxed{2} \begin{bmatrix} \textsc{synsem} & \begin{bmatrix} \textit{lex-synsem} & \\ \textsc{local}|\textsc{cat}|\textsc{head} & \boxed{1} \end{bmatrix} \end{bmatrix} \\ \textsc{non-head-dtr} & \boxed{3} \begin{bmatrix} \textsc{synsem}|\textsc{local}|\textsc{cat} & \begin{bmatrix} \textsc{head} & \begin{bmatrix} \textit{proper-noun} & \\ \textsc{name-sort} & \textit{anthroponym} \end{bmatrix} \\ \textsc{modification}|\textsc{modifiers} & \textit{no-modifiers} \end{bmatrix} \end{bmatrix} \\ \textsc{args} & \left\langle \boxed{2}, \boxed{3} \right\rangle \end{bmatrix}$$

Figure 8.7: Main syntactic constraints on *name-name* phrases.

proper-third-person-item, and simlutaneously the attribute NAME-SORT. In *noun-proper-second-or-third-person-item*, the relevant constraints are

$$\begin{bmatrix} \textit{noun-proper-second-or-third-person-item} & \\ \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} & \begin{bmatrix} \textsc{agr}|\textsc{person} & \textit{2nd-distant\_or\_3rd} \\ \textsc{name-sort} & \textit{anthroponym} \end{bmatrix} \end{bmatrix}$$

and they are

$$\begin{bmatrix} \textit{noun-proper-third-person-item} & \\ \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} & \begin{bmatrix} \textsc{agr}|\textsc{person} & \textit{3rd} \\ \textsc{name-sort} & \textit{not-anthroponym} \end{bmatrix} \end{bmatrix}$$

in *noun-proper-third-person-item.* The lexical types for proper names that allow second person readings inherit from *noun-proper-second-or-third-person-item*, and the ones that do not allow them inherit from *noun-proper-third-person-item.* The *name-name* phrase constrains both daughters to have a NAME-SORT of type *anthroponym.*

The key syntactic properties of the *name-name* phrase can thus be presented in Figure 8.7.

8.5.3.2 Semantics

If the piece of semantics a grammar associates to the proper name "João" is

$$\lambda x.named(x, \text{``}João\text{''})$$

and the one it associates to "Maria" is

$$\lambda x.named(x, \text{``}Maria\text{''})$$

then

$$\begin{aligned} &\lambda x.named(x, \text{``}João\text{''}) \wedge named(x, \text{``}Maria\text{''}) \\ &\wedge\ name\text{-}precedes(x, \text{``}João\text{''}, \text{``}Maria\text{''}) \end{aligned}$$

is an intuitive representation for the combination “João Maria”. The *name-precedes* relation is necessary because it does not follow from the fact that someone has the name “João” and the name “Maria” that their name is “João Maria” (it can be “Maria João”). The first argument of that *name-precedes* relation is necessary because the order can be different in different people’s names.

However, the LKB does not seem to allow a relation to have two arguments that are Lisp strings. The problem arises at generation (when the relations are indexed for generation, string literal arguments are also indexed, but apparently only the first one is considered). An ‘’invalid predicates” error message for the *name-precedes* relation appears and generation is aborted.

A fix to this problem is to use string variables. Instead of giving the expression “João Maria” the semantics presented above and repeated here

$$\begin{aligned}&\lambda x.named(x, \text{“}João\text{”}) \wedge named(x, \text{“}Maria\text{”})\\ &\wedge\ name\text{-}precedes(x, \text{“}João\text{”}, \text{“}Maria\text{”})\end{aligned}$$

we can produce

$$\begin{aligned}&\lambda x.\exists s_1.\exists s_2.named(x, s_1) \wedge string\text{-}equals(s_1, \text{“}João\text{”})\\ &\wedge\ named(x, s_2) \wedge string\text{-}equals(s_2, \text{“}Maria\text{”})\\ &\wedge\ name\text{-}precedes(x, s_1, s_2)\end{aligned}$$

instead. No relation now has two arguments that are Lisp strings, and the error is avoided.

If this semantics is to be produced in *name-name* phrases compositionally from the semantics of proper names, the latter must be changed. A proper name like “Maria” cannot come in the lexicon with the semantics

$$\lambda x.named(x, \text{“}Maria\text{”})$$

but rather with the much more verbose

$$\lambda x.\exists s_1.named(x, s_1) \wedge string\text{-}equals(s_1, \text{“}Maria\text{”})$$

as addressed in Section 8.5.1. This is because the second argument of the *named* relation cannot be changed from a string literal to a variable, since composition of semantics must be monotonic.

The actual implementation is as follows. First, the existential quantifiers in the formulas above are not explicitly included, but it is rather assumed that all string variables are existentially quantified.

Two subtypes of *string* are created: *string-literal* and *string-variable*. The latter also inherits from *individual*, because in the LinGO Grammar Matrix the ARG0 feature of relations is declared to be of this type, and some of the relations involved are implemented with an ARG0 of type *string-variable*.

In the LKB configuration file `globals.lsp` the parameter ***string-type*** is set to the new *string-literal*, effectively telling the LKB that TDL string literals (corresponding to Lisp strings) are accepted in all syntactic contexts where objects of type *string-literal* are accepted. In `mrsglobals.lisp` a condition is added to ***determine-variable-type*** associating *string-variable* instances to “s” (this makes *string-variable* instances be represented as s_n in MRSs, where n is an integer).

The MRS for the *name-name* phrase “Maria João” is presented in Figure 8.8. The *name-precedes* relation is contributed by the syntactic rule.

In Section 8.5.3.1 it was mentioned that only right recursive structures are accepted for semantic reasons. The explanation is that the ARG1 of the *name-precedes* relation comes from the left daughter and the ARG2 of the same relation comes from the right daughter (which in turn

$$\begin{bmatrix} \textit{mrs} & \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{name-precedes\_rel} & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x5} \\ \text{ARG1} & \boxed{s13}\ s \\ \text{ARG2} & \boxed{s12}\ s \end{bmatrix}, \begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{x5} \\ \text{ARG1} & \boxed{s13} \end{bmatrix}, \begin{bmatrix} \textit{string-equals\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{s13} \\ \text{CARG} & \textit{Maria} \end{bmatrix}, \begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{x5} \\ \text{ARG1} & \boxed{s12} \end{bmatrix}, \begin{bmatrix} \textit{string-equals\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{s12} \\ \text{CARG} & \textit{João} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 8.8: MRS fragment associated with the name-name phrase "Maria João"

comes from its own left daughter). For the phrase "João Sebastião Ribeiro", the right recursive binary structure [João [Sebastião Ribeiro]] produces a semantic representation that fully describes the ordering of the named involved. On the other hand, the left recursive structure [[João Sebastião] Ribeiro] would produce a semantic representation that only says that "João" precedes "Sebastião" and "João" precedes "Ribeiro", but does not say whether "Sebastião" precedes "Ribeiro" or "Ribeiro" precedes "Sebastião" (i.e. [[João Ribeiro] Sebastião] would be associated to equivalent semantics).

8.6 NP Structure

The table in Table 8.3 presents some of the data covered here.

The category Predeterminers (Position I) in this table contains elements like "todo" (*all*).

The category Determiners (Position II) includes the definite and indefinite articles, the demonstratives and other items, like "bastante(s)" (*much*, *several*).

The category Possessives (Position III) contains prenominal possessives, which in European Portuguese are preceded by determiners.

The category Cardinals (in Position IV) includes the cardinal numerals, either atomic ("dois", *two*) or complex ("vinte e dois", *twenty two*).

The category Ordinals (in Position IV) includes the ordinal numerals, atomic ("primeiro", *first*) and complex ones ("vigésimo primeiro", *twenty-first*).

The category Vague Quantifiers (in Position IV) contains elements like "muitos" (*many*), "poucos" (*few*). The distinction between determiners like "bastantes" and vague quantifiers is not semantic but syntactic. Consider their different behavior with respect to a preceding definite article, exemplified in the following sentences:

(42) a. [$_{\text{NP}}$ Muitas espécies de sapos da Amazónia] já estão extintas.
many species of frogs of the Amazon Rainforest already are extinct
Many species of frogs of the Amazon Rainforest are already extinct.

b. [$_{\text{NP}}$ Bastantes espécies de sapos da Amazónia] já estão
several species of frogs of the Amazon Rainforest already are
extintas.
extinct
Several species of frogs of the Amazon Rainforest are already extinct.

c. [$_{\text{NP}}$ As muitas espécies de sapos da Amazónia] já estão
the many species of frogs of the Amazon Rainforest already are
extintas.
extinct
The many species of frogs of the Amazon Rainforest are already extinct.

	I	II	III	IV	V	VI	VII	VIII	IX
(a)		a	minha	primeira		bicicleta		com pedais amarelos	
		the	*my*	*first*		*bicycle*		*with yellow pedals*	
	`my first bicycle with yellow pedals`								
(b)	todas	aquelas		três mil		pessoas		ali	
	all	*those*		*three thousand*		*people*		*over there*	
	`all those three thousand people over there`								
(c)		um		certo	grande	espírito			que criou o mundo
		a		*certain*	*great*	*spirit*			*that created the world*
	`a certain great spirit that created the world`								
(d)		a				invasão	americana	do Iraque	
		the				*invasion*	*American*	*of the Iraq*	
	`the American invasion of Iraq`								
(e)				quatro		colegas		teus	
				four		*collegues*		*of yours*	
	`four collegues of yours`								
(f)		a				pesca	baleeira	intensa	
		the				*fishing*	*whale-like*	*intense*	
	`the intense whale fishing`								
(g)		aquelas	suas	muitas		queixas			
		those	*their*	*many*		*complaints*			
	`those many complaints of theirs`								
(h)		o				papa		esse	que é tão snob
		the				*pope*		*that*	*who is such a snob*
	`that pope who is such a snob`								

Table 8.3: NP constituents. Positions within the Noun Phrase: I — Predeterminers; II — Determiners; III — Prenominal Possessives; IV — Cardinals (b) (e), Ordinals (a), Vague Quantifiers (g), Markers of Indefinite Specifics (c); V — Prenominal Adjective Phrases; VI — Head Noun; VII — Adjectival Arguments; VIII — Adjective Phrase Adjuncts (f), Prepositional Phrase Arguments (d), Prepositional Phrase Adjuncts (a), Adverbial Phrase Adjuncts (b), Postnominal Possessives (e), Postnominal Demonstratives (h); IX — Restrictive Relative Clauses.

d. * [$_{\text{NP}}$ As bastantes espécies de sapos da Amazónia] já estão
the several species of frogs of the Amazon Rainforest already are
extintas.
extinct

The category Indefinite Specifics (in Position IV) contains elements like "certo" and "determinado" (*certain*), that mark NPs with exclusively indefinite specific readings, as in the first example below (43a):

(43) a. Todas as pessoas leram um certo livro.
all the people have read a certain book
All people have read a certain book.
$\exists y[book(y) \wedge \forall x[person(x) \rightarrow read(x, y)]]$

b. Todas as pessoas leram um livro.
all the people have read a book
All people have read a book.
$\forall x[person(x) \rightarrow \exists y[book(y) \wedge read(x, y)]]$
$\exists y[book(y) \wedge \forall x[person(x) \rightarrow read(x, y)]]$

The most interesting property of these elements is that their contribution to the meaning of the sentences where they occur consists in merely restricting the relative scope possibilities between the quantifiers in these sentences. The example Portuguese sentence in (43b) is ambiguous between the two readings shown below it. In contrast, the example sentence in (43a) is not ambiguous and only has the reading where the existential quantifier has wide scope — its specific reading. This issue is explored in Section 8.6.4.1.

The category Prenominal Adjective Phrases (Position V) includes adjective phrases (APs) that precede the noun, and the slot named Head Noun (Position VI) represents the position where the noun surfaces.

The slot for Adjectival Arguments (Position VII) represents the position where adjectives that realize arguments of nouns surface. In the example in the table repeated below, the adjective form "americana" (*American*) realizes one of the arguments of the noun "invasão" (*invasion*). The semantics of this NP is quite similar to the semantics of a sentence like *The U.S. invaded Iraq*. More specifically, the arguments of the semantic relations for the noun "invasão"/*invasion* and the verb *invade* are the same in these examples.

(44) a invasão americana do Iraque
the invasion American of the Iraq
the American invasion of Iraq

In Position VIII one finds APs that do not saturate noun arguments, prepositional phrase (PP) adjuncts (not realizing noun arguments) and complements (realizing noun arguments), adverbial phrase (AdvP) adjuncts of nouns, postnominal demonstratives and postnominal possessives (adjuncts or complements). Not all adverbs can occur in this context (as noun modifiers). Among the adverbial phrases that can modify nouns one finds "aqui", "aí", "ali", "dentro (de NP)", "fora (de NP)", "junto (a/de NP)" respectively *here*, *there*, *there*, *inside (NP)*, *outside/out of NP*, *nearby/near NP*.

The last slot is for relative clauses (Position IX).

Elements occupying the same position in Table 8.3 generally show free word order among themselves (but, depending on the category of these elements, there are some restrictions that will be presented in the following Sections). For instance the relative word order between cardinals and ordinals (both in Position IV) is unconstrained:

(45) a. Os primeiros dois filmes foram cancelados.
the first two films were canceled
The first two films were canceled.

b. Os dois primeiros filmes foram cancelados.
the two first films were canceled
The two first films were canceled.

The numbering of these positions reflects precedence constraints among these elements: to give an example, prenominal adjectives (Position V) cannot precede cardinals (Position IV):

(46) a. Os adeptos entusiasmaram-se depois de [$_{\text{NP}}$ duas grandes vitórias do
the fans got excited after two great victories of the
clube.]
club
The fans got excited after two great victories of their club.

b. * Os adeptos entusiasmaram-se depois de [$_{\text{NP}}$ grandes duas vitórias do
the fans got excited after great two victories of the
clube.]
club

8.6.1 General Constraints

The type hierarchy in Figure 8.9 presents the values of the features MARKING and MARK that are used to account for the NP structure of Portuguese. The Head-Functor configurations presented in Section 5.3 are used to implements many of the NP constituents. This hierarchy controls their syntactic distribution, as explained in the following sections.

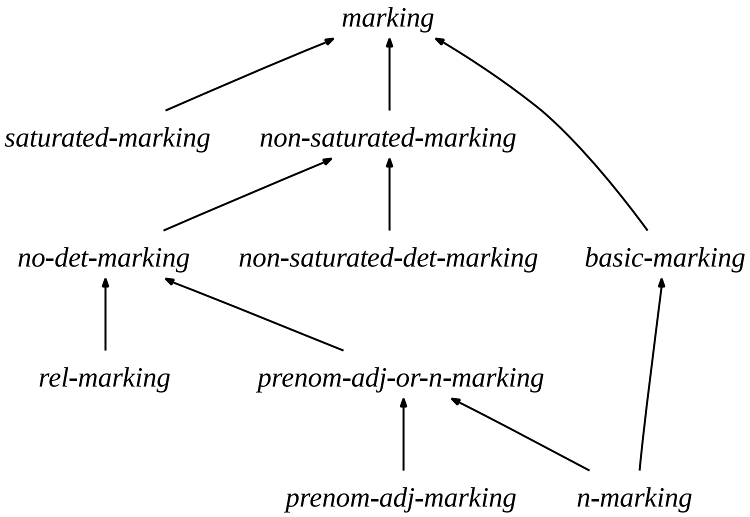


Figure 8.9: Simplified hierarchy for the values of the features MARKING and MARK in noun headed constituents.

Elements that select NPs select for a constituent with a feature MARKING of the type *saturated.* Nouns come in the lexicon with the type *n-marking* for this feature. Since these two types are incompatible according to this hierarchy, a noun needs to combine with other elements in order to form an NP.

The type *marking* is defined to have a subfeature MK-VAL, and that feature has the subfeatures DEMONSTRATIVE, QUALQUER, TAL, POSSESSIVE, OUTRO, CARDINAL, INDEF-SPEC and ORDINAL. These features are used to encode the presence of elements like possessives, cardinals, etc., and to control their iterability and co-occurrence. The feature DEMONSTRATIVE is for demonstratives, QUALQUER is for the element "qualquer" (*any*), TAL is for the prenominal "tal" (*such*), POSSESSIVE is for possessives, OUTRO is for the element "outro" (*other*), CARDINAL is for cardinals, ORDINAL is for ordinals, INDEF-SPEC is for prenominal "certo" and "determinado" (*certain*), and ORDINAL is for ordinals. They will also be explained in the next sections. The type hierarchy with the values that all these features can take is in Figure 8.10.

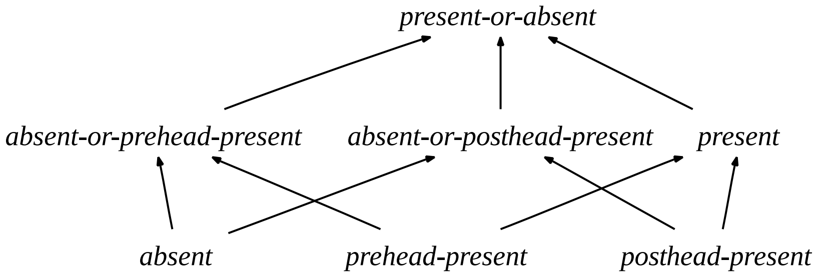


Figure 8.10: Type hierarchy under *present-or-absent.*

In this hierarchy, the type *absent* denotes the fact that the relevant element is not present in a given phrase. For instance, a constituent with this value for the feature POSSESSIVE is a constituent where no possessive occurs. The value *present* denotes the presence of the relevant element, and its two subtypes *prehead-present* and *posthead-present* state whether the relevant element precedes or follows the head noun respectively. Types like *absent-or-prehead-present* and *absent-or-posthead-present* are used to control co-occurrence restrictions between these elements.

The marking type *n-marking* is constrained in the following manner:

$$\begin{bmatrix} \textit{n-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{tal} & \textit{absent} \\ \textsc{possessive} & \textit{absent-or-posthead-present} \\ \textsc{cardinal} & \textit{absent} \\ \textsc{ordinal} & \textit{absent} \\ \textsc{indef-spec} & \textit{absent} \end{bmatrix} \end{bmatrix}$$

This type, *n-marking*, is the value of the feature MARKING of nouns. It is also the value of all noun headed constituents made up by a noun and postnominal material. Possessives, for instance, can follow the noun, so the value of the feature POSSESSIVE is *absent-or-posthead-present.* It will be *absent* for nouns (all the features under MK-VAL are constrained to be *absent* in the lexical entries of nouns), but it will take the value *posthead-present* if a postnominal possessive is present in a constituent with MARKING of the type *n-marking.*

Other elements that attach to noun-headed constituents (and whose presence is not represented with these attributes) simply unify the MK-VAL feature of the constituent that they select with the MK-VAL under their MARK feature. The example of adjectives illustrate this point They have the following constraints:

$$\begin{bmatrix} \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{marker} \begin{bmatrix} \textsc{mark}|\textsc{mk-val}\ \boxed{1} \\ \textsc{select}|\textsc{local}|\textsc{cat}|\textsc{marking}|\textsc{mk-val}\ \boxed{1} \end{bmatrix} \end{bmatrix}$$

Elements that select NPs also constrain them to have discharged COMPS, besides requiring them to have a MARKING feature compatible with the type *saturated-marking.* However, in LXGram this is not achieved by constraining the COMPS feature to be an empty list. Instead, this feature is constrained to be of a type similar to the type *olist* that comes in the LinGO Grammar Matrix. The name of this type is *list-of-optional-synsems* in LXGram. See Section 7.1.

8.6.2 Determiners and Predeterminers

Saturated NPs can be introduced by a determiner or a predeterminer:

(47) a. [$_{\text{NP}}$ Os$_{\text{D}}$ seres humanos] são livres.
the human beings are free

Human beings are free.

b. [$_{\text{NP}}$ Todos$_{\text{PreD}}$ os$_{\text{D}}$ seres humanos] são livres.
all the human beings are free

All human beings are free.

In the first case the quantifier relation of the NP comes from the determiner, but in the second case it comes from the predeterminer. When a predeterminer introduces an NP, a determiner must be present (48).

(48) a. todas as pessoas
all the people

all (the) people

b. todas aquelas pessoas
all those people

all those people

c. *EP/BP todas pessoas
all people

The last example is actually a possible NP in Brazilian Portuguese. More on this is said below.

Determiners that co-occur with predeterminers must thus be different from determiners introducing an NP, since the former contribute no quantifier semantics but the latter do. Multiple lexical items are required in view of the fact that it is not possible to underspecify the number of elementary predications that a given lexical item contributes to the MRS representation.

The lexical entries for determiners that contribute quantifier semantics and appear at the left edge of NPs (the form "os" in (47a)) are constrained in the following manner:

$$\begin{bmatrix} \text{SYNSEM}|\text{LOCAL} & \begin{bmatrix} \text{CAT}|\text{HEAD}|\text{MARKER} & \begin{bmatrix} \textit{pre-only-marker} & \\ \text{SELECT}|\text{LOCAL} & \begin{bmatrix} \text{CAT} & \begin{bmatrix}\text{HEAD} & \textit{noun}\\ \text{MARKING} & \textit{no-det-marking}\end{bmatrix} \\ \text{CONT}|\text{HOOK} & \begin{bmatrix}\text{LTOP} & \boxed{1}\\ \text{INDEX} & \boxed{2}\end{bmatrix}\end{bmatrix} \\ \text{MARK} & \textit{saturated-marking}\end{bmatrix} \\ \text{CONT} & \begin{bmatrix} \text{HOOK}|\text{LTOP} & \boxed{3} \\ \text{RELS} & \left\{\begin{bmatrix}\text{LBL} & \boxed{3}\\ \text{ARG0} & \boxed{2} \\ \text{RSTR} & \boxed{4}\end{bmatrix}\right\} \\ \text{HCONS} & \left\{\begin{bmatrix}\textit{qeq} & \\ \text{HARG} & \boxed{4}\\ \text{LARG} & \boxed{1}\end{bmatrix}\right\}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

The relevant constraints on the determiners that follow predeterminers and contribute no semantics (the form “os” in (47b)) are:

$$\begin{bmatrix} \text{SYNSEM}|\text{LOCAL} & \begin{bmatrix} \text{CAT}|\text{HEAD}|\text{MARKER} & \begin{bmatrix} \textit{pre-only-marker} & \\ \text{SELECT}|\text{LOCAL} & \begin{bmatrix} \text{CAT} & \begin{bmatrix}\text{HEAD} & \textit{noun}\\ \text{MARKING} & \textit{no-det-marking}\end{bmatrix} \\ \text{CONT}|\text{HOOK} & \boxed{1}\end{bmatrix} \\ \text{MARK} & \textit{non-saturated-det-marking}\end{bmatrix} \\ \text{CONT} & \begin{bmatrix} \text{HOOK} & \boxed{1} \\ \text{RELS} & \{\} \\ \text{HCONS} & \{\}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

They have the HOOK of their sister node (so that the LTOP of the mother node in Head-Functor rules, which comes from the functor daughter, is the same as the head daughter’s LTOP).

Predeterminers have constraints like:

$$\begin{bmatrix} \textsc{synsem}|\textsc{local} & \begin{bmatrix} \textsc{cat}|\textsc{head}|\textsc{marker} & \begin{bmatrix} \textit{pre-only-marker} & \\ \textsc{select}|\textsc{local} & \begin{bmatrix} \textsc{cat}|\textsc{head}\ \textit{noun} & \\ \textsc{cont}|\textsc{hook} & \begin{bmatrix} \textsc{ltop} & \boxed{1} \\ \textsc{index} & \boxed{2} \end{bmatrix} \end{bmatrix} \\ \textsc{prehead} & \begin{bmatrix} \textsc{select}|\textsc{local}|\textsc{cat}|\textsc{marking}\ \textit{non-saturated-det-marking} \\ \textsc{mark}\ \textit{saturated-marking} \end{bmatrix} \end{bmatrix} \\ \textsc{cont} & \begin{bmatrix} \textsc{hook}|\textsc{ltop} & \boxed{3} \\ \textsc{rels} & \left\{ \begin{bmatrix} \textsc{lbl} & \boxed{3} \\ \textsc{arg0} & \boxed{2} \\ \textsc{rstr} & \boxed{4} \end{bmatrix} \right\} \\ \textsc{hcons} & \left\{ \begin{bmatrix} \textit{qeq} & \\ \textsc{harg} & \boxed{4} \\ \textsc{larg} & \boxed{1} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

They require the presence of a semantically vacuous determiner (therefore they select for a sister node with MARKING of type *non-saturated-det-marking*). They produce a saturated phrase, since their feature MARK is of type *saturated-marking*. They also introduce quantifier semantics.

Predeterminers can also appear postnominally, as in (49). As the second example shows, they occupy an NP internal position. This means that syntactic and semantic scope do not match in such structures, and more features are therefore needed to pass the relevant information along the syntax tree. We will not elaborate on this issue, as this is left to future work (LXGram includes a preliminary treatment for postnominal universal quantifiers that, however, makes them outscope the rest of the NP and cannot account for the data in (49b)).

(49) a. as pessoas todas
the people all
all (the) people

b. as pessoas todas dessa aldeia
the people all from that village
all (the) people from that village

To account for Brazilian Portuguese “todo” (48c), we resort to positing more than one lexical entries for “todo”. The constraints associated with the HEAD attribute of this item only differ from the ones of the head type of predeterminers above in that, instead of selecting for a constituent with MARKING *non-saturated-det-marking*, this item selects for an element with MARKING *no-det-marking* (i.e. this item is encoded as a determiner).

8.6.3 Prenominal Possessives

In definite NPs, possessives can appear prenominally (50a), while postnominal possessives can occur in indefinite NPs (50b). However, demonstratives license both prenominal and postnominal possessives (51).

(50) a. A minha bicicleta tem um pneu furado.
the my bicycle has a tire flat
My bicycle has a flat tire.

b. Uma bicicleta minha tem um pneu furado.
a bicycle my/mine has a tire flat

A bicycle of mine has a flat tire.

(51) a. Aquela tua bicicleta tem um pneu furado.
that your bicycle has a tire flat

That bicycle of yours has a flat tire.

b. Aquela bicicleta tua tem um pneu furado.
that bicycle your/yours has a tire flat

That bicycle of yours has a flat tire.

Other contexts allow prenominal possessives. Examples are vocatives (52a) and predicative nominals lacking a determiner (52b).

(52) a. Minha senhora, eu quero a mala.
my lady, I want the bag

I'd like the bag, Miss.

b. É teu irmão?
is your brother

Is he your brother? (a brother of yours)

Postnominal possessives are covered in Section 8.6.11.2, as well as the mechanism to control relative word order between noun and possessive.

Prenominal possessives always occur after the determiner (article, demonstrative, . . .), if it is present, and they always precede cardinals, if both occur (53).

(53) a. as minhas duas bicicletas
the my two bicycles

my two bicycles

b. *minhas as duas bicicletas
my the two bicycles

c. *as duas minhas bicicletas
the two my bicycles

The HEAD of possessives is thus constrained in the following way:

$$\begin{bmatrix} \textit{possessive} & \\ \textsc{marker} & \begin{bmatrix} \textsc{select}|\textsc{local}|\textsc{cat} & \begin{bmatrix} \textsc{head} & \textit{noun} \\ \textsc{marking} & \begin{bmatrix} \textsc{mk-val} & \begin{bmatrix} \textsc{possessive} & \textit{absent} \\ \textsc{tal} & \boxed{1} \\ \textsc{cardinal} & \boxed{2} \\ \textsc{ordinal} & \boxed{3} \\ \textsc{indef-spec} & \boxed{4} \\ \textsc{outro} & \boxed{5} \\ \textsc{qualquer} & \boxed{6} \\ \textsc{demonstrative} & \boxed{7} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \textsc{mark} & \begin{bmatrix} \textsc{mk-val} & \begin{bmatrix} \textsc{possessive} & \textit{present} \\ \textsc{tal} & \boxed{1}\ \textit{present-or-absent} \\ \textsc{cardinal} & \boxed{2}\ \textit{present-or-absent} \\ \textsc{ordinal} & \boxed{3}\ \textit{present-or-absent} \\ \textsc{indef-spec} & \boxed{4}\ \textit{present-or-absent} \\ \textsc{outro} & \boxed{5}\ \textit{present-or-absent} \\ \textsc{qualquer} & \boxed{6}\ \textit{present-or-absent} \\ \textsc{demonstrative} & \boxed{7}\ \textit{absent-or-posthead-present} \end{bmatrix} \end{bmatrix} \\ \textsc{prehead} & \begin{bmatrix} \textsc{select}|\textsc{local}|\textsc{cat}|\textsc{marking}\ \textit{no-det-marking} & \\ \textsc{mark} & \begin{bmatrix} \textit{no-det-marking} & \\ \textsc{mk-val}|\textsc{possessive} & \textit{prehead-present} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

This accounts for prenominal possessives after a definite article or demonstrative determiner.

In Brazilian Portuguese, possessives can introduce an NP. In the corresponding phrases in European Portuguese the definite article must precede the possessive. A Brazilian example is in (54).

(54) Minha bicicleta tem um pneu furado.
my bicycle has a tire flat
My bicycle has a flat tire.

Since quantifier semantics is generally introduced in the predeterminer or determiner slots (Position I and Position II in Table 8.3), the Brazilian possessives must have different lexical items from the possessives occurring with determiners, because these do not carry quantifier semantics, but the former must do so (see the next section for semantic representations of possessives). Also, they will present different constraints related to MARKING. More specifically, the feature MARK bears the value *saturated*.

Also note that NPs introduced by a possessive, like the one in (54), do not have readings characteristic of bare NPs — but bare NPs headed by a singular count noun are actually possible in Brazilian Portuguese ([Munn and Schmitt, 1998] and [Müller, 2002]) —, so these NPs should not be considered to be bare NPs.

Prenominal possessives following a definite article or other determiner are also attested in Brazilian Portuguese.

This analysis also covers sequences made up by a predeterminer "todo" (Section 8.6.2) followed immediately by a possessive, which is a possibility in Brazilian Portuguese. These sequences are derived by the lexical entry for "todo" that is specific to Brazilian Portuguese and a lexical entry for a possessive that is available to both varieties.

8.6.3.1 Possessives as Arguments of Nouns

Possessives can realize arguments of noun relations, which in Portuguese are in the unmarked case realized by postnominal material (55a). Consider (55b).

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{x2}\ x \\ \text{RELS} & \left\langle \begin{matrix} \begin{bmatrix}\textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h1} \\ \text{ARG0} & \boxed{x2} \\ \text{RSTR} & \boxed{h4}\ h \\ \text{BODY} & \boxed{h3}\ h\end{bmatrix}, \begin{bmatrix}\textit{possessive\_a\_rel} & \\ \text{LBL} & \boxed{h5}\ h \\ \text{ARG0} & \boxed{e6}\ e \\ \text{ARG1} & \boxed{x2} \\ \text{ARG2} & \boxed{x7}\begin{bmatrix} x & \\ \text{PERSON} & \textit{3rd}\end{bmatrix}\end{bmatrix}, \\ \begin{bmatrix}\textit{pronoun\_q\_rel} & \\ \text{LBL} & \boxed{h8}\ h \\ \text{ARG0} & \boxed{x7} \\ \text{RSTR} & \boxed{h9}\ h \\ \text{BODY} & \boxed{h10}\ h\end{bmatrix}, \begin{bmatrix}\textit{pronoun\_n\_rel} & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x7}\end{bmatrix}, \begin{bmatrix}\textit{\_cavalo\_n\_rel} & \\ \text{LBL} & \boxed{h5} \\ \text{ARG0} & \boxed{x2}\end{bmatrix} \end{matrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix}\textit{qeq} & \\ \text{HARG} & \boxed{h4} \\ \text{LARG} & \boxed{h5}\end{bmatrix}, \begin{bmatrix}\textit{qeq} & \\ \text{HARG} & \boxed{h9} \\ \text{LARG} & \boxed{h11}\end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 8.11: MRS fragment corresponding to the NP “o seu cavalo” (*his/her/their horse*).

(55) a. o irmão da Ana
the brother of the Ana
Ana’s brother

b. o seu irmão
the her brother
her brother

In both examples, “irmão” denotes a two-place predicate. In (55a) the second argument is realized by the PP “de Ana”, and in (55b) it surfaces as “seu”.

Possessives are implemented in LXGram as carrying personal pronoun semantics (see Section 8.4).

When possessives do not fill a noun argument, an extra relation is included between the index of the personal pronoun and that of the head noun, called *possessive_a_rel*. An example is in Figure 8.11.

When possessives realize noun arguments, this relation is not present in the MRS. Instead, the index of the personal pronoun occurs as the second argument of the relation corresponding to the head noun. Figure 8.12 contains an MRS example of argumental possessives, for the NP “o seu irmão” (*his/her/their brother*).

Because the number of elementary predications contributed to an MRS by these two sorts of elements (argumental vs. modifying possessives) is different, multiple lexical entries are required for possessives.

Argumental possessives and modifying possessives have the same syntactic distribution, though. This creates problems for the treatment of argumental possessives, since we assume that noun complements are saturated at a much lower level (see Section 8.6.8).

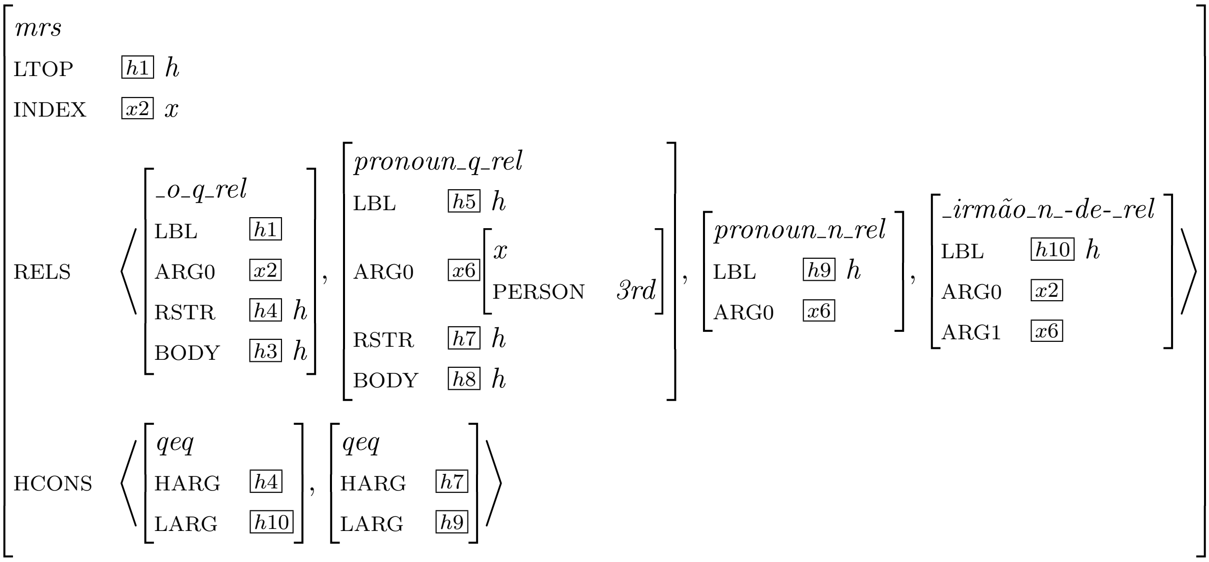


Figure 8.12: MRS fragment corresponding to the NP "o seu irmão" (*his/her/their brother*).

The first question to ask is whether projections of prenominal argumental possessives should be produced by some Head-Complement construction or by the *functor-head-phrase* discussed above.

The motivation for considering argumental possessives to be complements is that they are in complementary distribution with PP complements (56a). The motivation for considering them functors is that they are also in complementary distribution with modifying possessives (56b).

(56) a. * o seu irmão da Ana
the her brother of the Ana

b. * o seu seu irmão
the her her brother

If they are treated as functors, then they are unusual in saturating an argument of the head they select.

If they are complements, then prenominal argumental possessives are unusual in preceding the head (in Portuguese this only occurs with clitics and fronted constituents).

In LXGram they are implemented as functors. This choice has the advantage of not requiring more syntactic machinery, but it results in untypical feature structures because, since argumental possessives are considered functors, they cannot discharge an element from the COMPS list of their head, in spite of realizing it themselves.

Since they can see the entire SYNSEM of their sister node via the SELECT attributes, they can unify the index of the personal pronoun relations they introduce with the index of an element in the COMPS attribute of the nominal projection they select for. This produces the right semantics, namely semantic representations exactly like the ones produced by Head-Complement constructions.

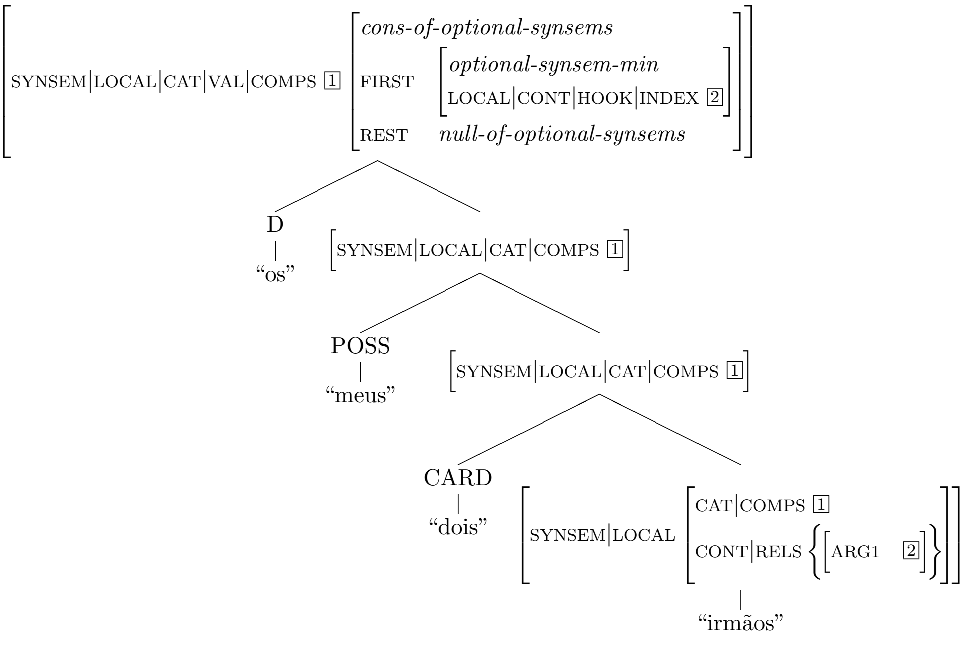
They place the same constraints on the values of *marking* as their modifier counterparts. Their non-iterability is in this way immediately predicted.

It is important to mention that what enables a non-empty COMPS to appear high enough in a tree in order to be visible by possessives in general is the choice of using the type *list-of-optional-synsems* instead of *null* to constrain the COMPS of NPs, as explained in Section 8.6.1. Consider the following example:

(57) [NP os [meus [dois irmãos]]]
the my two brothers
my two brothers

All bracketed phrases in this example have the same value for the feature COMPS according to the implementation. If NPs were constrained to have an empty COMPS, a unary rule would be needed to discharge the unexpressed complement of the noun (this rule could simply pass up the tail of the COMPS of its daughter). It would make sense to have this rule apply in the most embedded position (before the cardinal attaches), for several reasons: it is where Head-Complement constructions occur; some of the constraints common to unary and binary Head-Complement constructions could be factored out in a single supertype; discharging all complements in the same position is also less error-prone and makes the grammar easier to understand, to extend and to debug if needed. In this scenario, the sister node of the possessive would also have an empty COMPS.

By using the type *list-of-optional-synsems* to constrain the COMPS of NPs instead (Section 8.6.1), unrealized complements are visible at the point where possessives attach. For instance, the NP in (57) receives the following simplified analysis:



Relational nouns, like "irmão" (*brother*) above, unify the SARG (see Section 5.11) of their complement with the ARG1 of the relation they introduce, so that the entry for "irmão" contains these constraints, among others:

$$\begin{bmatrix} \text{SYNSEM}|\text{LOCAL} & \begin{bmatrix} \text{CAT}|\text{VAL}|\text{COMPS} & \left\langle \begin{bmatrix} \text{LOCAL}|\text{CONT}|\text{HOOK}|\text{SARG} & \boxed{1} \end{bmatrix} \right\rangle \\ \text{CONT}|\text{RELS} & \left\{ \begin{bmatrix} \text{ARG1} & \boxed{1} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Constraints in the lexical types for argumental possessives are used to unify the index associated with the personal pronoun relations that they introduce with the index of the first element in the head's COMPS:

$$\begin{bmatrix}\textsc{synsem}|\textsc{local} & \begin{bmatrix}\textsc{cat}|\textsc{head}|\textsc{marker}|\textsc{select}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{comps}|\textsc{first}|\textsc{local}|\textsc{cont}|\textsc{hook}|\textsc{sarg}\ \boxed{1}\\ \textsc{cont}|\textsc{rels}\left\{\begin{bmatrix}\textsc{arg0} & \boxed{1}\end{bmatrix}, \begin{bmatrix}\textsc{arg0} & \boxed{1}\end{bmatrix}\right\}\end{bmatrix}\end{bmatrix}$$

Crucially, the possessive cannot simply unify the entire SYNSEM of the head's complement with its own SYNSEM, for a number of reasons: (1) a cyclic structure would result, a situation that is not allowed by the systems used; (2) the noun selects for a PP, but a possessive is not a PP — the HEAD feature is different, for instance, and would not unify —; and (3) the first element of COMPS, which in examples like (57) is reduced to type *list-of-optional-synsems*, is an *optional-synsem-min*, but the SYNSEM of the possessive ends up as a *canonical-synsem-min*, since it is realized, and these synsem types are incompatible (see Section 8.6.1).

The fact that the complement of a noun with a synsem of type *optional-synsem-min* (the type of the SYNSEM of unexpressed elements) is actually realized makes this analysis rather uninteresting.

Since possessives can only realize PP complements of nouns (and not CPs for instance), argumental possessives must constrain the nominal projection they attach to to have a COMPS whose first element is a PP headed by a non-predicational preposition (see Section 7.7):

$$\begin{bmatrix}\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{marker}|\textsc{select}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{comps}|\textsc{first}|\textsc{local}|\textsc{cat}|\textsc{head}\ \textit{particle-np}\end{bmatrix}$$

This constraint, like the constraint above to fix the semantics, is extremely non-local and against the spirit of HPSG. Furthermore, it means that nouns do not necessarily have visibility over the entire SYNSEM of their complement: if nouns constrain it to be a PP, a possessive can detect this and realize it instead, but a possessive can have constrains on its SYNSEM drastically different from the constraints on the noun's complement. A consequence is that if constraints on noun complements must be added in the future to cover additional phenomena, it may be the case that the definitions for argumental possessives require modifications as well — the analysis is not extensible.

This is the analysis implemented in LXGram currently. An interesting alternative to the analysis of prenominal possessives would be to treat them as elements extracted from a postnominal position. An analysis could be envisaged in a way similar to the treatment of long-distance dependencies, but possibly resorting to other features, so as to not interact with the analysis of unbounded dependencies. This would explain the paradox of arguments realized by possessives surfacing on the left of their head, and, under the assumption of a parallelism between sentence structure and NP structure, it would provide the NP counterpart for the left periphery of sentences.

8.6.4 Cardinals, Ordinals, Vague Quantifiers, Markers of Indefinite Specific NPs

Position IV can be filled in by cardinals, ordinals or markers of indefinite specific NPs, like "certo" or "determinado" (*certain*).

They can co-occur with each other in almost any order (58), the exception being that ordinals cannot precede markers of indefinite specifics, as in (58d).

(58) a. os primeiros dois capítulos
the first two chapters
the first two chapters

b. os dois primeiros capítulos
the two first chapters
the first two chapters

c. um certo primeiro capítulo
a certain first chapter
a certain first chapter

d. * um primeiro certo capítulo
a first certain chapter
a certain first chapter

"Certo" is limited to indefinite NPs. Cardinals cannot co-occur with indefinite determiners,[2] so to test the word order possibilities between cardinals and "certo", we have to look at NPs that begin with a cardinal or "certo", as in (59). Such NPs are covered in Section 8.6.5, but (59) already shows that word order between cardinals and markers of indefinite specifics is in general also unconstrained.

(59) a. dois certos capítulos
two certain chapters
two certain chapters

b. certos dois capítulos
certain two chapters
two certain chapters

At most one item of each class can be present (60). They are not repeatable even when an item of a different sort intervenes (61).

(60) a. * Os dois três carros avariaram.
the two three cars broke down

b. * O primeiro segundo lugar está ocupado.
the first second seat is taken

c. * Um determinado certo carro avariou.
a certain certain car broke down

(61) a. * Os dois primeiros três lugares estão ocupados.
the two first three places are taken

b. * Os primeiros dois segundos pratos estão atrasados.
the first two second dishes are late

c. * Certos dois certos carros avariaram.
certain two certain cars broke down

A class of prenominals, "vague quantifiers" or "quantificational adjectives" (62), has the exact distribution of cardinals .

(62) Os vários participantes passeiam as folhas pela sala.
the various participants walk the paper sheets through the room
The various participants walk the paper sheets through the room.

They cannot co-occur with cardinals (63).

[2] NPs like *some three cars* can be analyzed as involving an item *some* that is not a determiner but rather a modifier of the cardinal, since *some three* roughly means *around three.* This also applies to Portuguese expressions like "alguns três", "uns três", with the same meaning.

(63) a. * os vários vinte participantes
the various twenty participants

b. * os vinte vários participantes
the twenty various participants

Vague quantifiers occur with ordinals (64).

(64) a. os vários primeiros lugares
the various first seats
the various first seats

b. os primeiros vários lugares
the first various seats
the various first seats

They cannot iterate (65).

(65) a. * os vários vários participantes
the various various participants

b. * os vários vinte vários participantes
the various vinte various participants

Vague quantifiers can thus be constrained exactly like cardinals. In the following discussion we will thus ignore them and only talk about cardinals. In LXGram they are implemented essentially as cardinals, with little differences.

Similarly, the class of ordinals can also be considered to include other elements with the same syntactic distribution. This is the case of items like "último" (*last*) and "próximo" (*next*). Consider:

(66) a. * os próximos primeiros capítulos
the next first chapters

b. * os primeiros próximos capítulos
the first next chapters

c. os três próximos capítulos
the three next chapters
the next three chapters

d. os próximos três capítulos
the next three chapters
the next three chapters

We will also have these elements in mind when we discuss ordinals, from now on. In LXGram these items receive the same lexical type as ordinals.

We now turn to the discussion of implementing these three classes: ordinals, cardinals/vague quantifiers and markers of indefinite specific NPs.

Cardinals have the following constraints under their HEAD:

$$\begin{bmatrix} \textit{cardinal} & \\ \textsc{marker} & \begin{bmatrix} \textit{pre-only-marker} & \\ \textsc{mark} & \begin{bmatrix} \textit{no-det-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{cardinal} & \textit{prehead-present} \\ \textsc{demonstrative} & \boxed{1} \\ \textsc{possessive} & \boxed{2} \\ \textsc{qualquer} & \boxed{3} \\ \textsc{tal} & \boxed{4} \\ \textsc{outro} & \boxed{5} \\ \textsc{indef-spec} & \boxed{6} \\ \textsc{ordinal} & \boxed{7} \end{bmatrix} \end{bmatrix} \\ \textsc{select}|\textsc{local}|\textsc{cat} & \begin{bmatrix} \textsc{head}\ \textit{noun} & \\ \textsc{marking} & \begin{bmatrix} \textit{no-det-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{cardinal} & \textit{absent} \\ \textsc{demonstrative} & \boxed{1}\ \textit{absent-or-posthead-present} \\ \textsc{possessive} & \boxed{2}\ \textit{absent-or-posthead-present} \\ \textsc{qualquer} & \boxed{3}\ \textit{present-or-absent} \\ \textsc{tal} & \boxed{4}\ \textit{present-or-absent} \\ \textsc{outro} & \boxed{5}\ \textit{present-or-absent} \\ \textsc{indef-spec} & \boxed{6}\ \textit{present-or-absent} \\ \textsc{ordinal} & \boxed{7}\ \textit{present-or-absent} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The HEAD of ordinals is:

$$\begin{bmatrix} \textit{ordinal} & \\ \textsc{marker} & \begin{bmatrix} \textit{pre-only-marker} & \\ \textsc{mark} & \begin{bmatrix} \textit{no-det-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{cardinal} & \boxed{1} \\ \textsc{demonstrative} & \boxed{2} \\ \textsc{possessive} & \boxed{3} \\ \textsc{qualquer} & \boxed{4} \\ \textsc{tal} & \boxed{5} \\ \textsc{outro} & \boxed{6} \\ \textsc{indef-spec} & \boxed{7} \\ \textsc{ordinal} & \textit{prehead-present} \end{bmatrix} \end{bmatrix} \\ \textsc{select}|\textsc{local}|\textsc{cat} & \begin{bmatrix} \textsc{head}\ \textit{noun} & \\ \textsc{marking} & \begin{bmatrix} \textit{no-det-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{cardinal} & \boxed{1}\ \textit{present-or-absent} \\ \textsc{demonstrative} & \boxed{2}\ \textit{absent-or-posthead-present} \\ \textsc{possessive} & \boxed{3}\ \textit{absent-or-posthead-present} \\ \textsc{qualquer} & \boxed{4}\ \textit{present-or-absent} \\ \textsc{tal} & \boxed{5}\ \textit{present-or-absent} \\ \textsc{outro} & \boxed{6}\ \textit{present-or-absent} \\ \textsc{indef-spec} & \boxed{7}\ \textit{absent} \\ \textsc{ordinal} & \textit{absent} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The constraint on the feature INDEF-SPEC to be *absent* is to prevent ordinals from preceding "certo" and "determinado", thus blocking examples like (58d).

The HEAD of markers of indefinite specifics is very similar:

$$\left[\begin{array}{ll}
\textit{indef-specific} & \\
\text{MARKER} & \left[\begin{array}{ll}
\textit{pre-only-marker} & \\
\text{MARK} & \left[\begin{array}{ll}
\textit{no-det-marking} & \\
\text{MK-VAL} & \left[\begin{array}{ll}
\text{CARDINAL} & \boxed{1} \\
\text{DEMONSTRATIVE} & \boxed{2} \\
\text{POSSESSIVE} & \boxed{3} \\
\text{QUALQUER} & \boxed{4} \\
\text{TAL} & \boxed{5} \\
\text{OUTRO} & \boxed{6} \\
\text{INDEF-SPEC} & \textit{present} \\
\text{ORDINAL} & \boxed{7}
\end{array}\right]
\end{array}\right] \\
\text{SELECT}|\text{LOCAL}|\text{CAT} & \left[\begin{array}{ll}
\text{HEAD} & \textit{noun} \\
\text{MARKING} & \left[\begin{array}{ll}
\textit{no-det-marking} & \\
\text{MK-VAL} & \left[\begin{array}{ll}
\text{CARDINAL} & \boxed{1}\ \textit{present-or-absent} \\
\text{DEMONSTRATIVE} & \boxed{2}\ \textit{absent} \\
\text{POSSESSIVE} & \boxed{3}\ \textit{absent-or-posthead-present} \\
\text{QUALQUER} & \boxed{4}\ \textit{present-or-absent} \\
\text{TAL} & \boxed{5}\ \textit{absent} \\
\text{OUTRO} & \boxed{6}\ \textit{present-or-absent} \\
\text{INDEF-SPEC} & \textit{absent} \\
\text{ORDINAL} & \boxed{7}\ \textit{present-or-absent}
\end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

The constraint on the feature DEMONSTRATIVE is because markers of indefinite specific NPs cannot co-occur with demonstratives in the same NP. Since prenominal demonstratives outscope markers of indefinite specifics, prenominal demonstratives also select for a sister node with the value *absent* for the feature INDEF-SPEC.

8.6.4.1 Semantics of Markers of Indefinite Specifics

Prenominal items like "certo" and "determinado" (*certain*) as in the examples (43), repeated below, carry no semantic relations but instead simply restrict the set of available readings:

(67) a. Todas as pessoas leram um certo livro.
all the people have read a certain book
All people have read a certain book.
$\exists y[book(y) \wedge \forall x[person(x) \rightarrow read(x,y)]]$

b. Todas as pessoas leram um livro.
all the people have read a book
All people have read a book.
$\forall x[person(x) \rightarrow \exists y[book(y) \wedge read(x,y)]]$
$\exists y[book(y) \wedge \forall x[person(x) \rightarrow read(x,y)]]$

The MRS assigned by LXGram to the sentence in (67a) is in Figure 8.13, and the MRS for the sentence in (67b) is in Figure 8.14.

The two MRSs have exactly the same relations and handle constraints. The only differences lie in the values of the features SCOPE in some of the handles (of type *h*) in these MRSs. The handles for which this feature is not displayed have it completely unconstrained (minimal types for handles are used to hide unconstrained SCOPE features).

Without these constraints for the SCOPE feature, these two MRSs can be scope resolved in the two following formulas:

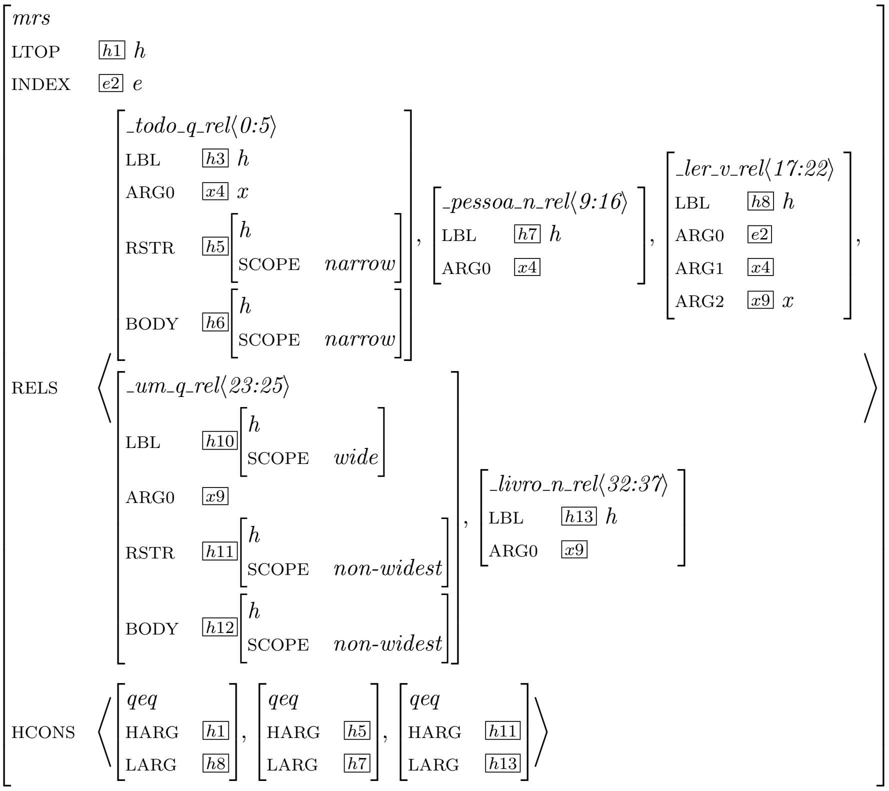
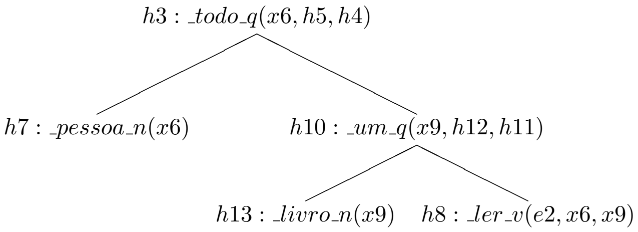


Figure 8.13: MRS with constrained quantifier scope. The sentence is “Todas as pessoas leram um certo livro” (*all people have read a certain book*).

- $\_um\_q(x9, \_livro\_n(x9), \_todo\_q(x6, \_pessoa\_n(x6), \_ler\_v(e2, x6, x9)))$
- $\_todo\_q(x6, \_pessoa\_n(x6), \_um\_q(x9, \_livro\_n(x9), \_ler\_v(e2, x6, x9)))$

These are in fact the two readings for the sentence in (67b). The constraints on SCOPE are intended to block the second reading for the example (67a), in Figure 8.13.

The idea is that, in the second reading, the handle tagged with $\boxed{h4}$ and the handle tagged with $\boxed{h10}$ in these MRSs correspond to the same node in the syntax tree for the scoped formula:



Since they represent the same node, we can assume that they must be compatible. The approach is then to make the constraints on these two handles incompatible in the MRS for the example (67a), but compatible in the MRS for the sentence in (67b).

The type hierarchy for the values that the feature SCOPE can take is in Figure 8.15.

The MRS in Figure 8.13 (for the example in (67a)) has $\boxed{h4}$ with its feature SCOPE with the value *narrow*, and the SCOPE feature of the handle $\boxed{h10}$ has the value *wide*. These types are incompatible according to the hierarchy in Figure 8.15.

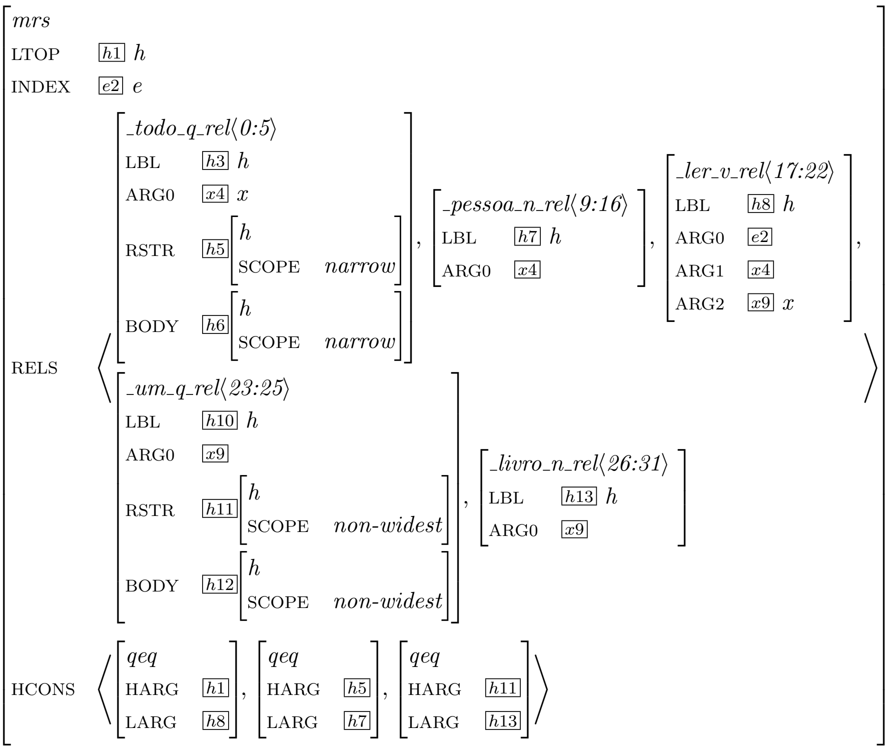


Figure 8.14: MRS allowing for quantifier scope ambiguity. The sentence is "Todas as pessoas leram um livro" (*all people have read a book*).

This mechanism does not work in practice, because the LKB scope resolution algorithm does not perform unification operations on the handles that end up denoting the same node in the fully scoped formulas. For this reason, these constraints on handles are still experimental in LXGram. It would of course be possible to resolve MRSs with an external component, that could take this information into account.

Markers of indefinite specifics contribute no semantics. Instead they simply constrain the features SCOPE of the associated quantifier relation. In LXGram, the quantifier relation of an NP is accessible in all noun headed phrases in the feature QUANT-REL under a feature KEYS (for key relations).[3]. Markers of indefinite specific NPs simply have the constraint:

$$\left[\text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{HEAD}|\text{MARKER}|\text{SELECT}|\text{LOCAL}|\text{CONT}|\text{KEYS}|\text{QUANT-REL}|\text{LBL}|\text{SCOPE}\ \textit{wide}\right]$$

If a marker of indefinite specifics is not present, this feature will simply have a more general type (allowing for more scope resolution possibilities, as desired).

The type *widest* in Figure 8.15 is used to constrain the LBL of the *proper_q_rel*, which is used with proper names. It is meant to ensure that proper names receive widest scope. The value *wide* is given to the LBL of quantifier relations associated with an NP where a marker of indefinite specifics is present. The RSTR and BODY features of all quantifier relations are constrained with the SCOPE value *non-widest*, except in *proper_q_rel* relation, where they are not constrained. Quantifier relations of determiners and predeterminers that cannot occur with markers of indefinite specifics in the same NP (e.g. "todo" — *all*) have the SCOPE under these

[3] The items that introduce quantifier relations simply unify this relation with the value of their head's QUANT-REL feature. The feature KEYS is unified between the mother node and the head daughter in all headed constructions.

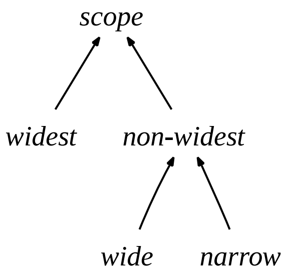


Figure 8.15: Type hierarchy under *scope*

two features (RSTR and BODY) further constrained to be *narrow.* This set of items contains the predeterminer "todo" (*all*), the definite articles and the demonstratives:

(68) a. * (Todos) os determinados homens leram um livro.
all the certain men have read a book

b. * Esses determinados homens leram um livro.
those certain men have read a book

The scope ordering is thus the following: *widest* > *wide* > *narrow*.
Consider the following example:

(69) Todos os filhos da Ana leram um certo livro.
all the children of the Ana have read a certain book
All of Ana's children have read a certain book.

These constraints license only one quantifier scope possibility:

$$proper\_q(x8, named(x, \text{``}Ana''), \_um\_q(x16, \_livro\_n(x16), \_todo\_q(x4, \_filho\_n\_ - de - (x4, x8), \_ler\_v(e2, x4, x16))))$$

The *proper_q_rel* relation cannot be embedded under any of the other quantifier relations, because its LBL has SCOPE of the type *widest*, but the handle arguments of the other quantifier relations have the value *non-widest* or *narrow* for their SCOPE features, and any of these types is incompatible with the type *widest.* The *_um_q_rel* relation in this example also cannot be under the scope of the *_todo_q_rel*, like in the previous example.

8.6.5 Cardinals and Markers of Indefinite Specifics as Determiners

As was mentioned in Section 8.6.4, cardinals and items like "certo" and "determinado" (*certain*) that mark indefinite specific NPs can themselves introduce an NP. Some examples from (59) are repeated below in (70).

(70) a. dois certos capítulos
two certain chapters
two certain chapters

b. certos dois capítulos
certain two chapters
two certain chapters

Ordinals cannot occur in NP initial position, though:

(71) a. um DVD com dois primeiros episódios dessa série
a DVD with two first episodes of that series
a DVD with two first episodes of that series

b. * um DVD com primeiros dois episódios dessa série
a DVD with first two episodes of that series

When these elements are preceded by a determiner, it is the determiner that introduces quantifier semantics. Assuming that quantifier semantics is always introduced by a determiner or a bare NP construction, there are two ways of introducing quantifiers in these NPs: considering them instances of bare NPs or analyzing the first element as a determiner.

The first possibility is not very attractive for Portuguese, for a number of factors. Preverbal bare NP subjects have a very constrained distribution in European Portuguese. It is interesting to note that NPs introduced by a cardinal (72c) do not pattern with bare NPs (72a) in the following examples, but rather with NPs introduced by determiners (72b).

(72) a. */?? Cartas chegaram.
letters have arrived

b. Algumas cartas chegaram.
some letters arrived
Some letters arrived.

c. Duas cartas chegaram.
two letters arrived
Two letters arrived.

The example in (72c) sounds as good as the one in (72b), which is introduced by the item "alguns", which can only occur in NP initial position and is thus not a bare NP.

Second, bare NPs tend to have non-specific readings in Portuguese: they cannot scope over negation (73), universal quantifiers (74) or intensional verbs (75). These examples are Brazilian Portuguese, from [Munn and Schmitt, 1998] (corresponding logical formulas added for ease of exposition), but the same observations hold for European Portuguese. We also bracketed the relevant bare NPs in these examples.

(73) a. João não viu uma mancha no chão.
João not saw a spot on the floor
João didn't see a spot on the floor.

1. $\neg\exists x[spot\_on\_the\_floor(x) \wedge saw(Jo\tilde{a}o, x)]$
2. $\exists x[spot\_on\_the\_floor(x) \wedge \neg saw(Jo\tilde{a}o, x)]$

b. João não viu [manchas no chão.]
João not saw spots on the floor
João didn't see spots on the floor.

1. $\neg\exists x[spot\_on\_the\_floor(x) \wedge saw(Jo\tilde{a}o, x)]$

(74) a. Todo mundo leu um livro sobre girafas.
everyone read a book on giraffes
Everyone read a book on giraffes.

1. $\forall x[person(x) \rightarrow \exists y[book\_on\_giraffes(y) \wedge read(x, y)]]$
2. $\exists y[book\_on\_giraffes(y) \wedge \forall x[person(x) \rightarrow read(x, y)]]$

b. Todo mundo leu [livros sobre girafas.]
everyone read books on giraffes
Everyone read books on giraffes.
1. $\forall x[person(x) \rightarrow \exists y[book\_on\_giraffes(y) \wedge read(x, y)]]$

(75) a. Pedro quer encontrar um policial.
Pedro wants to meet a policeman
Pedro wants to meet a policeman.
1. $\exists x[policeman(x) \wedge want(Pedro, meet(Pedro, x))]$
2. $want(Pedro, \exists x[policeman(x) \wedge meet(Pedro, x)])$

b. Pedro quer encontrar [policiais.]
Pedro wants to meet policemen
Pedro wants to meet policemen.
1. $want(Pedro, \exists x[policeman(x) \wedge meet(Pedro, x)])$

NPs introduced by cardinals do not pattern with bare NPs in this respect and allow both readings. An example with negation is in (76), in which the semantics of the cardinal "duas"/*two* is represented by $\lambda P.\lambda Q.\exists x_1 \exists x_2[x_1 \neq x_2 \wedge P(x_1) \wedge P(x_2) \wedge Q(x_1) \wedge Q(x_2)]$. Ambiguity can be found in the other two contexts as well.

(76) João não viu duas manchas no chão.
João not saw two spots on the floor
João didn't see two spots on the floor.
1. $\neg\exists x_1 \exists x_2[x_1 \neq x_2 \wedge spot\_on\_the\_floor(x_1) \wedge spot\_on\_the\_floor(x_2)$
$\wedge saw(João, x_1) \wedge saw(João, x_2)]$
2. $\exists x_1 \exists x_2[x_1 \neq x_2 \wedge spot\_on\_the\_floor(x_1) \wedge spot\_on\_the\_floor(x_2)$
$\wedge \neg saw(João, x_1) \wedge \neg saw(João, x_2)]$

Furthermore, NPs introduced by markers of indefinite specifics should obviously not be analyzed as bare NPs if the latter are constrained to take non-specific readings:

(77) a. O João não viu certa mancha no chão.
the João not saw certain spot on the floor
João didn't see a certain spot on the floor.
1. $\exists x[spot\_on\_the\_floor(x) \wedge \neg saw(João, x)]$

b. Todas as pessoas leram certo livro sobre girafas.
all the people read certain book on giraffes
Everyone read a certain book on giraffes.
1. $\exists y[book\_on\_giraffes(y) \wedge \forall x[person(x) \rightarrow read(x, y)]]$

c. Pedro quer encontrar certo polícia.
Pedro wants to meet certain policeman
Pedro wants to meet a certain policeman.
1. $\exists x[policeman(x) \wedge want(Pedro, meet(Pedro, x))]$

Bare NPs do not co-occur with the "cada" (*each*) of (78), but NPs introduced by cardinals do (examples from [Müller, 2002]):

(78) a. Os países da UE mandaram um/dois/vários delegado(s) cada.
the countries of the EU sent a/two/various delegate(s) each
The EU countries sent a/two/various delegate(s) each.

b. * Os países da UE mandaram delegados cada.
the countries of the EU sent delegates each

We conclude that cardinals and markers of indefinite specifics at NP initial position are best treated as determiners. They introduce indefinite NPs and cannot co-occur with prenominal possessives. The constraints on the *marking* features must therefore be different from the ones on elements of Position IV. Therefore, the constraints on their HEAD must differ. NP initial cardinals also carry quantifier semantics, which the elements of Position IV arguably do not. We will be calling them cardinal determiners and indefinite specific determiners from now on.

The HEAD of an NP initial cardinal looks like this:

$$\begin{bmatrix} \textit{cardinal-det} \\ \text{MARKER} \begin{bmatrix} \textit{pre-only-marker} \\ \text{MARK} \begin{bmatrix} \textit{saturated-marking} \\ \text{MK-VAL} \begin{bmatrix} \text{CARDINAL} & \textit{present} \\ \text{DEMONSTRATIVE} & \boxed{1} \\ \text{POSSESSIVE} & \boxed{2} \\ \text{QUALQUER} & \boxed{3} \\ \text{TAL} & \boxed{4} \\ \text{OUTRO} & \boxed{5} \\ \text{INDEF-SPEC} & \boxed{6} \\ \text{ORDINAL} & \boxed{7} \end{bmatrix} \end{bmatrix} \\ \text{SELECT}|\text{LOCAL}|\text{CAT} \begin{bmatrix} \text{HEAD } \textit{noun} \\ \text{MARKING} \begin{bmatrix} \textit{no-det-marking} \\ \text{MK-VAL} \begin{bmatrix} \text{CARDINAL} & \textit{absent} \\ \text{DEMONSTRATIVE} & \boxed{1}\ \textit{absent} \\ \text{POSSESSIVE} & \boxed{2}\ \textit{absent-or-posthead-present} \\ \text{QUALQUER} & \boxed{3}\ \textit{present-or-absent} \\ \text{TAL} & \boxed{4}\ \textit{present-or-absent} \\ \text{OUTRO} & \boxed{5}\ \textit{present-or-absent} \\ \text{INDEF-SPEC} & \boxed{6}\ \textit{present-or-absent} \\ \text{ORDINAL} & \boxed{7}\ \textit{present-or-absent} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The constraint on MK-VAL|CARDINAL of the selected synsem prevents cardinal determiners from iterating and from combining with the elements of Position IV.

The constraints for indefinite specific determiners are similar, with the obvious differences regarding the features CARDINAL and INDEF-SPEC.

There are no determiner versions of ordinals, as they cannot initiate an NP.

There are two questions to address: the relation between these determiners and the items of Position IV, and preventing bare NPs from being formed from NPs starting with an element in Position IV.

There are two possibilities for the first issue: to produce the determiner version from the postdeterminer one via a unary rule, or to have multiple lexical entries. In LXGram indefinite specifics receive multiple lexical entries, but cardinals do not. This is for reasons related to the composition of semantics and is explained in Section 8.6.5.1.

In LXGram bare NPs are produced by a unary syntactic rule that adds quantifier semantics, imposes a value of marking on the mother node subsumed by *saturated-marking* and requires the

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{matrix} \begin{bmatrix} \textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h6}\ h \\ \text{BODY} & \boxed{h5}\ h \end{bmatrix}, \begin{bmatrix} \textit{cardinal\_rel} & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{e9}\ e \\ \text{ARG1} & \boxed{h8}\ h \\ \text{ARG2} & \boxed{j10}\ j \end{bmatrix}, \begin{bmatrix} \textit{greater-or-equal\_rel} & \\ \text{LBL} & \boxed{h7} \\ \text{ARG0} & \boxed{j10} \\ \text{ARG1} & \boxed{j11}\ j \end{bmatrix}, \\ \begin{bmatrix} \textit{int-equals\_rel} & \\ \text{LBL} & \boxed{h7} \\ \text{ARG0} & \boxed{j11} \\ \text{CARG} & \textit{2} \end{bmatrix}, \begin{bmatrix} \textit{\_carro\_n\_rel} & \\ \text{LBL} & \boxed{h8} \\ \text{ARG0} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_avariar\_v\_rel} & \\ \text{LBL} & \boxed{h12} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \end{matrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h12} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h6} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 8.16: MRS of a sentence with a postdeterminer cardinal. The sentence is "os dois carros avariaram" (*the two cars broke down*).

daughter to be a noun headed sign with a MARKING subsumed by *no-det-marking* (see Section 8.8). In order to prevent bare NPs to be built from constituents that include a postdeterminer cardinal, ordinal or marker of indefinite specifics, the daughter is also constrained to have the features ORDINAL, CARDINAL and INDEF-SPEC under MARKING|MK-VAl of type *absent*.

8.6.5.1 Cardinal Determiners and the Semantics of Cardinals

In LXGram we chose to relate cardinal determiners and cardinal postdeterminers via unary syntactic rules. In particular, the determiner versions are produced from the postdeterminer versions. This is tied to issues of composition of semantics.

For the postdeterminer cardinals, an example of the MRSs produced is in Figure 8.16. The cardinal corresponds to three relations in this MRS: the *cardinal_rel* relation, the *greater-or-equal_rel* relation, and the *int-equals_rel* relation.

In the literature, there are several approaches to the semantics of cardinals: they have been given the semantic types $\langle e,t\rangle$ (a set of entities),[4] $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ (a function from sets to sets) or $\langle\langle e,t\rangle,\langle\langle e,t\rangle,t\rangle\rangle$ (a determiner). We did not choose to give cardinals quantifier semantics, because they can occur after determiners, as in expressions like *all three*. When they do appear in NP initial position, quantifier semantics must be added, though. We opted for the $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ treatment (i.e. consider them modifiers), and do not commit to saying that cardinals are intersective modifiers.[5] Therefore the *cardinal_rel* scopes over the relation introduced

[4] Or rather its characteristic function, a function from entities to truth values yielding true for all members of that set and for them only, i.e. a function of the form $\lambda x_{\in D_e}.P(x)$.

[5] We do not commit to saying that this function is $\lambda P_{\in D_{\langle e,t\rangle}}.\lambda x_{\in D_e}.P(x) \wedge Q(x)$, for some lexically given set Q. For example, if the denotation of *car* is $\lambda x_{\in D_e}.car'(x)$, the denotation of *two cars* would be $(\lambda P_{\in D_{\langle e,t\rangle}}.\lambda x_{\in D_e}.P(x) \wedge 2(x))(\lambda x_{\in D_e} car'(x)) = \lambda x_{\in D_e}.car'(x) \wedge 2(x)$ if intersective semantics is given to cardinals. This only makes sense if we consider the existence of plural (non-atomic) entities, whose atoms can be counted, in which case the set denoted by 2 in the above formula is the set of all plural entities with two atoms. There are many views on the semantics of cardinals, and we remain neutral with respect to the status of plural entities. In [Ionin and Matushansky, 2006]

by the head noun in MRSs. This is compatible with intersective semantics but does not enforce it. Our representation for cardinals is similar to $\lambda i_{\in I}.\lambda P_{\in D_{\langle e,t\rangle}}.\lambda x_{\in D_e}.cardinal(e, P(x), i)$, where the integer argument i is supplied in each lexical entry for cardinals. However, we do not define the meaning of the *cardinal* relation. It can be intersective if we posit that $\lambda i.\lambda P.\lambda x.cardinal(e, P(x), i) = \lambda i.\lambda P.\lambda x.P(x) \wedge count(P, i)$, where $\lambda i.\lambda x.count(x, i)$ is true if x is a plural entity with i atoms. Its meaning can however be defined differently, not necessarily in an intersective way.

Note that a definition that constrains the cardinality of the set denoted by the noun does not work. For instance a sentence like *three cars broke down* does not mean that the cardinality of the set of cars is three, but rather that the cardinality of the intersection of the set of cars and the set of things that broke down is three. Using plural entities, this sentence would be assigned a representation that says that there is a plural entity consisting of three cars that also belongs to the set of things that broke down, i.e. it is simple existential quantification, which is the semantics we will assume for cardinals occupying a determiner position (see below).

The other relations describe the integer argument of the *cardinal_rel* relation. It is widely assumed that an expression like *two children* means *at least two children* and not *exactly two children.* In the following example, the answer would be contradictory if *two children* meant *exactly two children*:

(79) — Do you have two children?
— Yes. In fact I have three.

The relevant piece of semantics is $cardinal(e_9, \_carro\_n(x_4), j_{10}) \wedge greater\text{-}or\text{-}equal(j_{10}, j_{11}) \wedge int\text{-}equals(j_{11}, 2)$. Variables of the form jn, where $n > 0$ are integer variables.[6] We can view these integer variables as existentially quantified by convention, so we do not explicitly include these quantifiers in the MRSs.

It would be more simple to produce $cardinal(e9, \_carro\_n(x4), 2)$, assuming that the relation greater or equal is part of the meaning of *cardinal_rel*.

The motivation for introducing the *greater-or-equal_rel* relation is that in certain contexts we do not want it to appear in the MRSs. This is the case of expressions like "exactly two" or "at most two". For an expression like "no máximo dois"/*at most two*, we can think of the semantics $\lambda P.\lambda x.cardinal(e, P(x4), j1) \wedge less\text{-}or\text{-}equal(j1, j2) \wedge int\text{-}equals\_rel(j2, 2)$. In order to factor out the similarity with the representation for an unmodified "dois"/*two*, we explicitly introduce *greater-or-equal_rel* relation when a cardinal is not modified.[7]

The use of the *int-equals_rel* relation is a matter of convenience. It is not necessary, because, instead of the piece of semantics $cardinal(e9, \_carro\_n(x4), j10) \wedge greater\text{-}or\text{-}equal(j10, j11) \wedge int\text{-}equals(j11, 2)$, we could simply use $cardinal(e9, \_carro\_n(x4), j10) \wedge greater\text{-}or\text{-}equal(j10, 2)$.

It is more convenient for the generation algorithm in the LKB to associate at least one relation with every lexical item. If we did not include this relation in the lexical entry for cardinals, their only semantic content would be the integer constant that is an argument of relations like *greater-or-equal_rel* or *less-or-equal_rel.* The implementation in LXGram associates to lexical items for cardinals only the *int-equals_rel* relations. All other relations are introduced in syntax, via rules that add semantics.

there are references to the main pieces of work in this field.

[6] They can be created by manipulating the LKB configuration files, namely by redefining the function **determine-variable-type** in the file `mrsglobals.lisp`.

[7] If we assumed that the *greater-or-equal_rel* relation is part of the meaning of *cardinal_rel* so that we would not have to include it in MRSs when a cardinal is not modified, expressions like "at most two" would not receive the correct semantics, or we could not use the *cardinal_rel* in these cases.

The first set of rules allows these expressions to combine with cardinal modifiers like *exactly*, *at most*, etc. Only one modifier is allowed, and if no modifier is present, a unary syntactic rule is used to add the *greater-or-equal_rel*. A cardinal modifier like *at most* introduces a *less-or-equal_rel*, a modifier like *exactly* introduces no relation.

Immediately up the tree, a unary rule is used to add the *cardinal_rel* relation and producing a node with a HEAD of type *cardinal*. After this step the piece of semantics for "dois" (*two*) and for "pelo menos dois" (*at least two*) is like $\lambda P.\lambda x.cardinal(e, P(x), j_1) \wedge greater\text{-}or\text{-}equal(j_1, j_2) \wedge int\text{-}equals(j_2, 2)$. The semantics for "no máximo dois" (*at most two*) is like $\lambda P.\lambda x.cardinal(e, P(x), j_1) \wedge less\text{-}or\text{-}equal(j_1, j_2) \wedge int\text{-}equals(j_2, 2)$. The semantics for "exactamente dois" (*exactly two*) is like $\lambda P.\lambda x.cardinal(e, P(x), j_1) \wedge int\text{-}equals(j_1, 2)$.

An optional rule can apply afterwards, changing the postdeterminer cardinal into a determiner (*cardinal-det* above) and adding quantifier semantics. So cardinal determiners are produced from cardinal postdeterminers via a syntactic rule.

We will not show the details of all these rules since they are relatively trivial. To control order of rule application LXGram uses different values of HEAD for these elements: many of these rules are non-headed. Only the two highest rules create nodes with values of *head* that inherit from *functor* and that can attach to nominal projections. These subtypes of *head* and their definitions (*cardinal* and *cardinal-det*) have already been presented.

This analysis is completely monotonic: we only add relations to an MRS, never remove or alter relations introduced elsewhere. This is a requirement of the LKB: composition of semantics has to be monotonic so that efficient algorithms can be used for generation. Also, every lexical entry for cardinals and every rule used in this process contributes at least one relation to the MRS.

Although a large number of dedicated rules is involved, they are used to build the semantics little by little and factor out the commonalities between the various pieces of MRS that are related to cardinals.

We assume that complex cardinal expressions like "vinte e um"/*twenty one* are recognized by a Named Entity Recognizer (NER) in a preprocessing step, and for the purposes of the grammar behave just like atomic cardinals like "vinte"/*twenty*. There is one NER developed in the University of Lisbon [Ferreira et al., 2007], that can be integrated with LXGram. Since NERs are not necessarily bidirectional, we can parse these expressions but we cannot generate them so far.

8.6.6 Modifying Adjectives

On a first approximation, adjectives select for a constituent with [MARKING *n-marking*] and produce a node with the same level of saturation:

$$\begin{bmatrix} \textit{adjective} & \\ \text{MARKER} & \begin{bmatrix} \text{SELECT|LOCAL|CAT} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{MARKING} & \textit{n-marking} \end{bmatrix} \\ \text{MARK } \textit{n-marking} & \end{bmatrix} \end{bmatrix}$$

As a consequence, they are allowed to recur.

Portuguese has prenominal and postnominal adjectives. Potentially spurious attachment ambiguities will be produced for a sequence AP$_1$-Noun-AP$_2$: [AP$_1$ [Noun AP$_2$]] and [[AP$_1$ Noun] AP$_1$]. Although spurious ambiguity is innocuous, it is also a source of inefficiency, as it causes the parser to perform more computations than needed. In LXGram the type hierarchy of *marking* is used to control this, too.

Examples like the one in (80) argue in favor of the structure [AP$_{1}$ [Noun AP$_{2}$]], since this NP can describe someone who is not Chinese. Accordingly, we want to provide to such NP semantics like $\lambda P.\_um\_q(x, \_falso\_a(e_{1}, \_médico\_n(x) \wedge \_chinês\_a(e_{2}, x)), P(x))$.[8] It does not describe a Chinese person who is a fake doctor (i.e. $\lambda P.\_um\_q(x, \_falso\_a(e_{1}, \_médico\_n(x)) \wedge \_chinês\_a(e_{2}, x), P(x)))$. Assuming syntactic scope and semantic scope match, the structure [AP$_{1}$ [Noun AP$_{2}$]] is justified.[9]

(80) um falso médico chinês
a fake doctor Chinese
a fake Chinese doctor

Prenominal adjectives are specified to have the constraint [MARK *prenom-adj-marking*] and select for nominal projections with [MARKING *prenom-adj-or-n-marking*], while postnominal adjectives select for sister nodes with [MARKING *n-marking*] and also bear the value *n-marking* for their MARK attribute. The type of HEAD in adjectives has the following constraints:

$$\begin{bmatrix} \textit{adjective} & \\ \text{MARKER} & \begin{bmatrix} \text{SELECT|LOCAL|CAT} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{MARKING} & \textit{prenom-adj-or-n-marking} \end{bmatrix} \\ \text{MARK } \textit{prenom-adj-or-n-marking} & \end{bmatrix} \end{bmatrix}$$

In the lexical types for adjectives, we distinguish between the adjectives that can only precede the noun, the ones that can only follow it and the ones that can occur in either position. The following examples illustrate these three classes. An adjective like "mero" (*mere*) can only precede the noun, an adjective like "japonês" (*Japanese*) can only follow the noun, and an adjective like "falso" (*false*) can precede or follow it.

(81) a. Atacaram um mero inspector.
they attacked a mere inspector
They attacked a mere inspector.

b. * Atacaram um inspector mero.
they attacked an inspector mere

c. * Atacaram um japonês inspector.
they attacked a Japanese inspector

d. Atacaram um inspector japonês.
they attacked an inspector Japanese
They attacked a Japanese inspector.

e. Atacaram um falso inspector.
they attacked a false inspector
They attacked a false inspector.

f. Atacaram um inspector falso.
they attacked an inspector false
They attacked a false inspector.

[8] We can assume that the semantic representation of "falso" (*fake*), $\lambda P_{\in D_{\langle e,t\rangle}}.\lambda x_{\in D_{e}}.\_falso\_a(e, P(x))$, means $\lambda P.\lambda x.\neg P(x)$.

[9] It is not required that syntactic and semantic scope match, because it is possible to manipulate feature structures, but it is desirable that they do, since implementation becomes more straightforward if they match. We thus assume that syntax and semantics match in the absence of a compelling argument against it.

The lexical types for the adjectives that can precede the noun have the constraints:

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}\begin{bmatrix}\textit{adjective}\\ \textsc{marker}|\textsc{prehead}|\textsc{mark}\ \textit{prenom-adj-marking}\end{bmatrix}\right]$$

The lexical types for the ones that can follow the noun are constrained with:

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}\begin{bmatrix}\textit{adjective}\\ \textsc{marker}|\textsc{posthead}\begin{bmatrix}\textsc{select}|\textsc{local}|\textsc{cat}|\textsc{marking}\ \textit{n-marking}\\ \textsc{mark}\ \textit{n-marking}\end{bmatrix}\end{bmatrix}\right]$$

The adjectives that can follow or precede the noun inherit all these constraints. The ones that can only precede it are given a lexical type that inherits from the type where the constraints on PREHEAD are stated and is further constrained with:

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{marker}\ \textit{pre-only-marker}\right]$$

Likewise, the lexical type for the adjectives that can only follow the noun inherits from the type above that has a constrained POSTHEAD feature and is defined to also bear:

$$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{marker}\ \textit{post-only-marker}\right]$$

In this implementation, nouns are given the same syntactic distribution as noun-adjective sequences: they can combine with another adjective to their right, or with a prenominal adjective. Nouns have a syntactic distribution different from adjective-noun sequences, as the latter cannot combine with an adjective to their right.

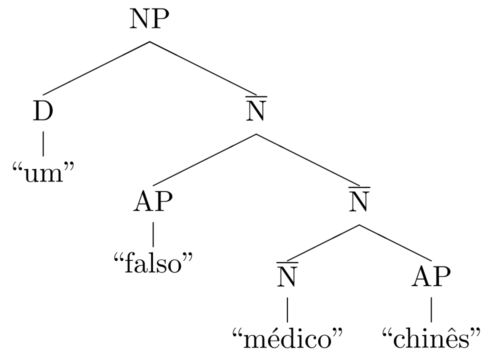
With this system of constraints, the noun phrase "um médico chinês falso" receives a semantic representation equivalent to "um falso médico chinês", equivalent to the lambda formula presented above. On the other hand, a noun phrase like "um médico falso Chinês" (*a fake doctor who is Chinese*) receives semantics equal to $\lambda P.\_um\_q(x, \_falso\_a(e_1, \_médico\_n(x)) \wedge \_chinês\_a(e_2, x), P(x))$, based on the syntactic structure ["um" [["médico falso"] "chinês"]].

Adjectives are allowed to iterate in both positions (prenominal and postnominal). This is borne out by data like:

(82) a. Era um grande, grande filme.
it was a great great movie
It was a great, great movie.

b. Era um filme chato, chato.
it was a movie boring boring
It as a boring, boring movie.

It is worth pointing out that we cannot properly capture the meaning difference between an $\overline{\text{N}}$ like "filme chato" (*boring movie*), which receives semantics equivalent to $\lambda x.\_filme\_n(x) \wedge \_chato\_a(e1, x)$, and an $\overline{\text{N}}$ like "filme chato, chato" (*boring, boring movie*), which is assigned an MRS representation equivalent to $\lambda x.\_filme\_n(x) \wedge \_chato\_a(e1, x) \wedge \_chato\_a(e2, x)$: the two formulas are logically equivalent due to idempotence of conjunction if we ignore the different event variables. It is not clear that the difference is truly semantic, anyway. It may simply be a pragmatic effect.



$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{functor-head-phrase}}\\
\textsc{stem} & \left\langle \text{“um”, “falso”, “médico”, “chinês”} \right\rangle\\
\textsc{synsem}|\textsc{local}|\textsc{cat} & \left[\begin{array}{lll}\textsc{head} & \boxed{1} & \textit{noun}\\ \textsc{marking} & \boxed{2} & \textit{saturated-marking}\end{array}\right]\\
\textsc{non-head-dtr} & \left[\begin{array}{ll}\textsc{stem} & \left\langle \text{“um”} \right\rangle\\ \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{determiner}}\\ \textsc{marker} & \left[\begin{array}{ll}\textsc{select} & \boxed{3}\\ \textsc{mark} & \boxed{2}\end{array}\right]\end{array}\right]\end{array}\right]\\
\textsc{head-dtr} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{functor-head-phrase}}\\
\textsc{stem} & \left\langle \text{“falso”, “médico”, “chinês”} \right\rangle\\
\textsc{synsem}\ \boxed{3} & \left[\textsc{local}|\textsc{cat}\left[\begin{array}{lll}\textsc{head} & \boxed{1} & \\ \textsc{marking} & \boxed{4} & \textit{prenom-adj-marking}\end{array}\right]\right]\\
\textsc{non-head-dtr} & \left[\begin{array}{ll}\textsc{stem} & \left\langle \text{“falso”} \right\rangle\\ \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{adjective}}\\ \textsc{marker} & \left[\begin{array}{ll}\textsc{select} & \boxed{5}\\ \textsc{prehead} & \left[\begin{array}{ll}\textsc{select} & \boxed{5}\\ \textsc{mark} & \boxed{4}\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right]\\
\textsc{head-dtr} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{head-functor-phrase}}\\
\textsc{stem} & \left\langle \text{“médico”, “chinês”} \right\rangle\\
\textsc{synsem}\ \boxed{5} & \left[\textsc{local}|\textsc{cat}\left[\begin{array}{lll}\textsc{head} & \boxed{1} & \\ \textsc{marking} & \boxed{6} & \textit{n-marking}\end{array}\right]\right]\\
\textsc{non-head-dtr} & \left[\begin{array}{ll}\textsc{stem} & \left\langle \text{“chinês”} \right\rangle\\ \textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{head} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{adjective}}\\ \textsc{marker} & \left[\begin{array}{ll}\textsc{select} & \boxed{7}\\ \textsc{posthead} & \left[\begin{array}{ll}\textsc{select} & \boxed{7}\\ \textsc{mark} & \boxed{6}\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right]\\
\textsc{head-dtr} & \left[\begin{array}{ll}\textsc{stem} & \left\langle \text{“médico”} \right\rangle\\ \textsc{synsem}\ \boxed{7} & \left[\textsc{local}|\textsc{cat}\left[\begin{array}{ll}\textsc{head} & \boxed{1}\\ \textsc{marking} & \textit{n-marking}\end{array}\right]\right]\end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

Figure 8.17: Syntactic analysis for an NP with a prenominal and a postnominal adjective

The syntactic analysis produced by LXGram for the NP in (80) ("um falso médico chinês" — *a false Chinese doctor*) is in Figure 8.17, with abridged feature structures.

The structure [[AP$_{\mathit{1}}$ N] AP$_{\mathit{2}}$] is blocked, because the phrase with the form [AP$_{\mathit{1}}$ N] has MARKING with the value *prenom-adj-marking* but postnominal adjectives select for a sister node with the value *n-marking* for that feature. There is no unifier for *n-marking* and *prenom-adj-marking*, as can be seen in Figure 8.9.

8.6.7 Argumental Adjectives

Semantics

Adjectives that are used as arguments of nouns (83a) display drastically different semantics from adjectives that modify a noun (83b). Consider the two examples:

(83) a. Viram [$_{\text{NP}}$ a alunagem americana] na televisão.
they saw the moon landing American on the television
They saw the American moon landing on TV.

b. Viram [$_{\text{NP}}$ um carro americano] naquela rua.
they saw a car American on that street
They saw an American car on that street.

The NP in the first example has semantics quite similar to an NP like "a alunagem pelos americanos" ("the moon landing of the Americans"). The semantics for this NP could be

$$\lambda P_{\in D_{\langle e,t\rangle}}.\_o\_q(x, \_o\_q(y, \_americano\_n(y), \_alunagem\_n(x,y)), P(x))$$

For the NP in the first example ("a alunagem americana" — *the American moon landing*) we could thus think of similar semantics. The semantics for the noun "alunagem" is

$$\lambda \mathcal{Q}_{\in D_{\langle\langle e,t\rangle t\rangle}}.\lambda x_{\in D_e}.\mathcal{Q}(\lambda y_{\in D_e}.alunagem(x,y))$$

Assuming that semantically, the noun is the functor and the adjective is the argument, the semantics for the argumental adjective in (83a) would have to be

$$\lambda P_{\in D_{\langle e,t\rangle}}.\_o\_q(z, \_americano\_n(z), P(z))$$

The most simple semantics for the modifying adjective "americano" in (83b) is

$$\lambda P_{\in D_{\langle e,t\rangle}}.\lambda x_{\in D_e}.P(x) \wedge \_americano\_a(e,x)$$

The semantics for the NP in (83b) is thus

$$\lambda P_{\in D_{\langle e,t\rangle}}.\_um\_q(x, \_carro\_n(x) \wedge \_americano\_a(e,x), P(x))$$

The same adjective in these two contexts presents very different semantics. There are two options: to have multiple lexical entries for the adjectives that can occur as modifiers and as arguments; to use an optional lexical rule to change the meaning and syntactic properties of such adjectives, producing one of the versions from the other, which would be in the lexicon.

The lexical rule approach is certainly more appealing, since adjectives that can occur as arguments would simply receive a special lexical type in their lexical entry, denoting this property. The problem is that we cannot produce one of the semantic representations from the other with the machinery in the LKB, because we cannot manipulate strings, and the mapping between the relation names *_americano_n* and *_americano_a* requires string manipulation.

In the current implementation of LXGram, multiple lexical entries are provided for these adjectives. For instance, for the adjective "americano" (*American*) there is one lexical entry for the case when the adjective has modifier semantics and another lexical entry for the situations when it has the semantics of a noun phrase.

Note that argumental adjectives can realize any noun argument when they combine with nouns that have several arguments. For example, the noun "invasão" (*invasion*) is defined in the lexicon has having two arguments, one corresponding to the subject of the verb *invade* and the other to its complement. In LXGram two lexical rules control this. The two rules are *arg-adj-1st_comp* and *arg-adj-2nd_comp-rule*. The first rule is used to obtain the readings where the adjective realizes the first argument of the noun, and the second rule for the readings where the adjective realizes its second argument. Argumental adjectives are forced by the implementation to always undergo one of these rules.

Syntax

A PP complement cannot intervene between a noun and an adjectival complement:

(84) a. a invasão americana do Iraque
the invasion American of the Iraq
The American invasion of Iraq

b. * a invasão do Iraque americana
the invasion of the Iraq American

Also, the remaining elements in Position VIII cannot appear before an adjective argument either. An example with a PP adjunct follows:

(85) a. a alunagem americana de 1969
the moon landing American of 1969
the American moon landing of 1969

b. * a alunagem de 1969 americana
the moon landing of 1969 American

In LXGram, there are two important subtypes of *synsem*s: *lex-synsem* and *phrase-synsem*. The SYNSEM feature of all words (terminal symbols and lexical rules) is of the type *lex-synsem*, and the SYNSEM of phrases is of the type *phrase-synsem*. These types are incompatible: they have no common subtype.

Argumental adjectives are implemented as functors. They select a sister node that has a SYNSEM of the type *lex-synsem* (i.e. the SELECT feature of argumental adjectives is constrained to be a *lex-synsem*).

Because of this constraint, nothing can intervene between the noun and this type of adjective, because, if that happened, a phrasal node would have to be the head daughter of this construction. This constraint also has the nice side effect of blocking two adjectival arguments of the same noun. This blocks the following ungrammatical example:

(86) a. * a invasão americana iraquiana
the invasion American Iraqi

The noun complements that are realized by adjectives are not discharged from the valence lists of nouns. This is similar to the implementation of argumental possessives (see Section 8.6.3.1). It is therefore necessary to ensure that a possessive and an adjective do not end up picking up the same argument of the head noun.

Although the noun complements that are realized by possessives and adjectives are not removed from the noun's valence lists, they are marked to have a SYNSEM with the type *optional-synsem*. This is the type for unexpressed elements, and this constraint is enough to ensure that these complements are not projected via the head-complement constructions.

There are two incompatible subtypes of *optional-synsem* in LXGram: *realized-by-a-possessive* and *realized-by-an-adjective*. Argumental adjectives constrain the complement of the noun that they realize to be of the type *realized-by-an-adjective*, and possessives constrain it to be of the type *realized-by-a-possessive*. The possibility of the same noun complement to be associated with both a possessive and an adjective in the semantic representations is blocked because these two types have no common subtype.

8.6.8 Noun Complementation

Many nouns subcategorize for one or more complements, that can be of different kinds. For the sake of illustration, here we will focus only on nouns with a single PP complement.

The standard HPSG approach to project complements is assumed: subcategorized for complements are members of a list-valued attribute COMPS in the lexical entry of the corresponding head, and a syntactic rule projects elements in that list, producing a mother node with a reduced COMPS.

Following many computationally implemented HPSGs, like the LinGO English Resource Grammar or the LinGO Grammar Matrix, strict binary branching is assumed — in the case of multiple complements, they are discharged one at a time. The Head-Complement syntactic rule or rules therefore unify the SYNSEM of the non-head daughter with the first element in the COMPS of the head daughter, and the COMPS of the mother node is the tail of the COMPS of the head daughter.

An issue in focus here is the relative scope between complements and the various functors. In Portuguese, the relative order between complements and several adnominal constituents (the ones in Position VIII in Table 8.3) is free. Consider the examples in (87).

(87) a. o consumo galopante$_{AP}$ [$_{PP}$ de petróleo]
the consumption ever increasing of oil
the ever increasing consumption of oil

b. o consumo [$_{PP}$ de petróleo] galopante$_{AP}$
the consumption of oil ever increasing
the ever increasing consumption of oil

These examples show that word order between postnominal adjunct adjectives and PP complements is arbitrary. Similar data can be presented for the other elements in Position VIII.

With other functors, however, word order is not free. Indeed, PP complements surface before restrictive relative clauses (but see Section 11.2):

(88) a. o consumo [$_{PP}$ de petróleo] [$_{RelCl}$ que continua a crescer]
the consumption of oil that continues to increase
the consumption of oil that continues to increase

b. *o consumo [$_{RelCl}$ que continua a crescer] [$_{PP}$ de petróleo]
the consumption that continues to grow of oil

The exact constraints on the position of relative clauses within an NP, as they are implemented in LXGram, are presented in Section 11.2.

Since complements occupy the same word order slot as the functors that give rise to constituents with *n-marking*, the relative syntactic scope between complements and the remaining functors must be the same as the relative scope between *n-marking* functors and the rest.

If complement placement is not constrained, many attachment ambiguities will surface. There will be no corresponding differences in the semantics produced, because the semantic constraints that link the MRS representation for the noun and the MRS representation for its complements are completely lexical (given in the lexical entries for nouns) and not affected by syntax.

There are two possible solutions to prevent this spurious overgeneration. The first one is to have all functors except the ones that occur in Position VIII select for a projection with saturated COMPS. Since the functors that occur in Position VIII are the most deeply embedded ones, if a complement is projected, it can only occur also in this position.

The second solution involves constraining the value of MARKING in the mother node of Head-Complement rules to be of the type *basic-marking* (see Figure 8.9; *basic-marking* is a supertype of other values of *marking*, not shown in that figure, for phrases not headed by a noun). For instance, if the head daughter of a Head-Complement constructions is a phrase introduced by a cardinal, that phrase will have a MARKING value that is the unifier of the type *no-det-marking* (the type of the feature SYNSEM|LOCAL|CAT|HEAD|MARKER|MARK of cardinals; see Section 8.6.4) with the type *basic-marking* (this is the constraint on Head-Complement rules just mentioned). This unifier is the type *n-marking* accoording to the hierarchy in Figure 8.9. This type *n-marking* is defined to have a feature MK-VAL|CARDINAL with the value *absent* (see Section 8.6.1). However the value for this feature will be *present* for such a phrase, because of the constraints on cardinals (cardinals have the value *present* for their feature SYNSEM|LOCAL|CAT|HEAD|MARKER|MARK, as presented in Section 8.6.4). The types *absent* and *present* are incompatible (Figure 8.10). This shows how cardinals are prevented to attach lower than complements. The control on the attachment position of the other functors with respect to complements is similar. In particular, all functors with a MARK feature that includes constraints incompatible with the constraints on the type *n-marking* are forced to attach higher than complements.

In either case, Head-Complement rules unify the MARKING value of the head daughter with the MARKING value of the mother node:

$$\begin{bmatrix} \text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{MARKING } \boxed{1} \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOCAL}|\text{CAT}|\text{MARKING } \boxed{1} \end{bmatrix}$$

The second solution has an important advantage over the first one: the level of saturation at which complements attach is stated in a single place, a type for Head-Complement constructions. This is the solution used in LXGram. Note that *basic-marking* results in *n-marking* when unified with any type under *no-det-marking* (except for *prenom-adj-marking* and *rel-marking*, which are never selected for by a functor; see Figure 8.9).

Figure 8.18 shows the analysis of an NP with an adjective intervening between the head and the complement. The NP is “um membro provável do IRA” (*a probable member of the IRA*).

In this example the node labeled $\overline{\text{N}}$ is produced via a Head-Complement construction. The remaining phrasal nodes are produced via Head-Functor constructions.

The semantic representation for this NP produced by LXGram is equivalent to

$$\lambda P_{\in D_{\langle e,t\rangle}}.proper\_q(x_1, named(x_1, \text{“}IRA''), \_um\_q(x_2, \_provável(e, \_membro\_n(x_2, x_1)), P(x_2)))$$

8.6.9 PPs and AdvPs

PPs and some AdvPs can modify a noun on their left. Some examples are given in (89).

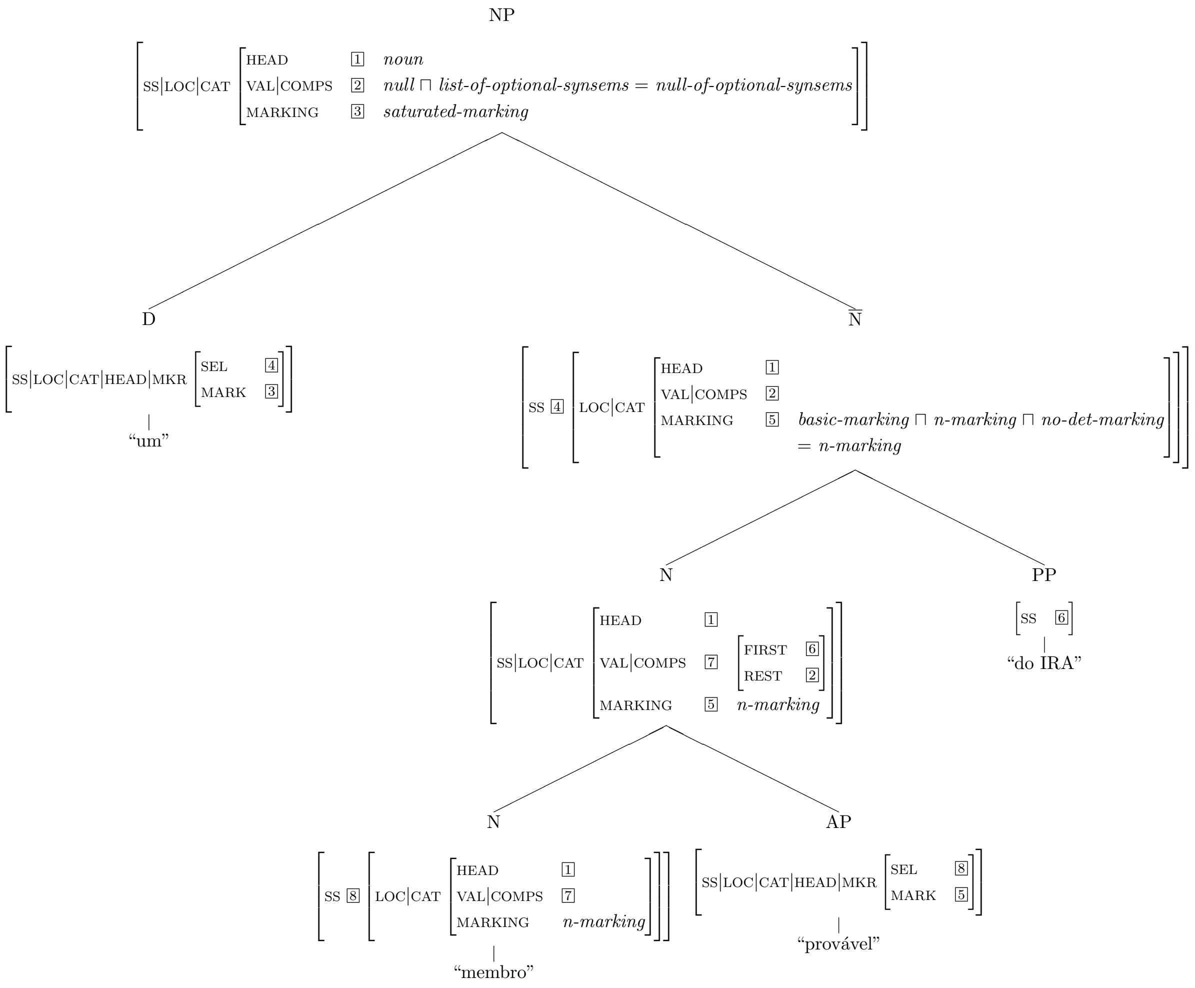


Figure 8.18: Syntactic analysis for "um membro provável do IRA" (*a probable member of the IRA*). SS abbreviates SYNSEM, LOC abbreviates LOCAL, MKR abbreviates MARKER, and SEL abbreviates SELECT.

(89) a. pessoas [$_{PP}$ com mobilidade reduzida]
people with mobility reduced
people with reduced mobility

b. [$_{NP}$ Aquele carro ali$_{AdvP}$ que tem um pneu furado] [$_{VP}$ estava estacionado aqui
that car there that has a tire flat was parked here
ontem.]
yesterday
That car over there that has a flat tire was parked here yesterday.

The adverb "ali" (*there*) in (89b) must be analyzed as part of the subject of the main clause, since it occurs in an NP internal position (following the noun "carro" and preceding the relative clause).

PPs and AdvPs cannot precede the noun, as shown in (90).

(90) a. * [$_{PP}$ com mobilidade reduzida] pessoas
with mobility reduced people

b. * Aquele ali$_{AdvP}$ carro estava estacionado aqui ontem.
that there car was parked here yesterday

Inside the NP, they have the syntactic distribution of postnominal adjectives (Position VIII in Table 8.3). In fact, because PPs and APs can be interspersed, ambiguity can arise concerning adjective attachment — consider (91).

(91) carros sem assentos vermelhos
cars without seats red
red cars with no seats/cars without red seats

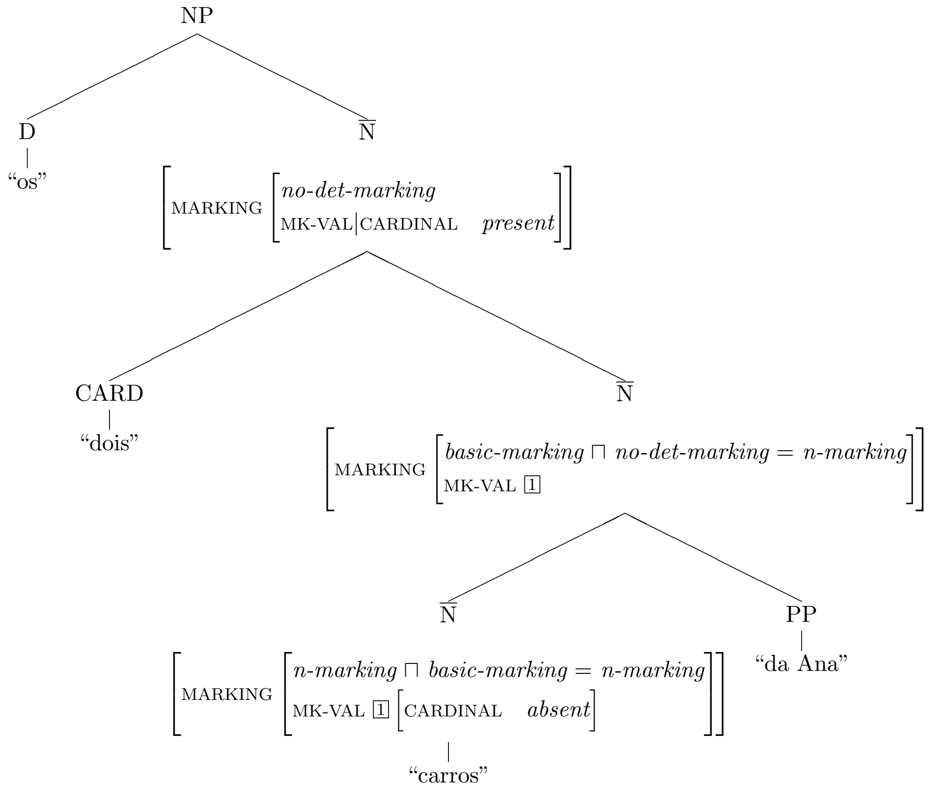
This example has two interpretations depending on the attachment site of the adjective: [[carros [$_{PP}$ sem assentos]] vermelhos$_{AP}$] (*red cars with no seats*) and [carros [$_{PP}$ sem assentos vermelhos]] (*cars without red seats*). This fact leads one to posit constraints on the head types of prepositions and adverbs similar to the ones on postnominal adjectives, as far as the feature POSTHEAD is concerned.

The constraints on the head of prepositions and adverbs that can modify nouns (as well as verbs) thus look like:

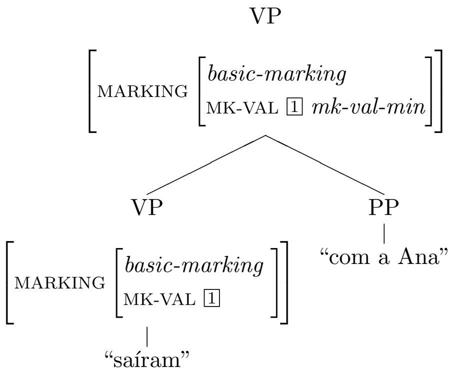
$$\left[\begin{array}{ll}\text{MARKER} & \left[\begin{array}{ll}\text{MARK} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{basic-marking}}\\ \text{MK-VAL} & \boxed{1}\end{array}\right]\\ \text{SELECT}|\text{LOCAL}|\text{CAT} & \left[\begin{array}{ll}\text{HEAD} & \textit{noun-or-verb}\\ \text{MARKING} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{basic-marking}}\\ \text{MK-VAL} & \boxed{1}\end{array}\right]\end{array}\right]\\ \multicolumn{2}{l}{\text{PREHEAD}|\text{SELECT}|\text{LOCAL}|\text{CAT}|\text{HEAD}\ \textit{verb}}\end{array}\right]\end{array}\right]$$

The analysis for the NP in (92) is shown below.

(92) os dois carros da Ana
the two cars of the Ana
Ana's two cars



The types subsumed by *no-det-marking* are not used to constrain the MARKING of verbal projections, because the items where they are employed only attach to nouns. This means that when a PP attaches to a verbal constituent, the MARKING value of that constituent remains *basic-marking*. The result is never *n-marking*. This way, features under MK-VAL like CARDINAL are not present in verb headed elements. Consider the analysis for the VP "saíram com a Ana" (*left with Ana*):



The general type *basic-marking* helps in hiding the implementation of NP structure. The information about the exact position within the NP where PPs and AdvPs attach — namely the constraints on the absence of the items in Position IV — is encapsulated in a more specific type, *n-marking*. This more specific type is kept separate from the definitions of prepositions and adverbs.

8.6.10 Relative Clauses

In LXGram, relative clauses are not headed by a verb, and a dedicated head type is used for them (*relative-complementizer*), i.e. constructions that project relative pronouns to the left of a clause are assumed to be non-headed structures. This is compatible with the LinGO Grammar Matrix. We abstain from developing on the analysis of relative clauses here, as their implementation is still experimental.

There is a type in the hierarchy under *marking*, *rel-marking*, used to model relative clause attachment.

Restrictive relative clauses should be allowed to iterate, but they are more peripheral than APs and PPs inside an NP, and they always follow the noun:

$$\begin{bmatrix} \textit{relative-complementizer} & \\ \text{MARKER} & \begin{bmatrix} \textit{post-only-marker} & \\ \text{SELECT}|\text{LOCAL}|\text{CAT} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{MARKING} & \begin{bmatrix} \textit{no-det-marking} \\ \text{MK-VAL}\ \boxed{1} \end{bmatrix} \end{bmatrix} \\ \text{MARK} & \begin{bmatrix} \textit{rel-marking} \\ \text{MK-VAL}\ \boxed{1} \end{bmatrix} & \end{bmatrix} \end{bmatrix}$$

The type hierarchy under *marking* and the constraints presented so far mean that restrictive relative clauses outscope prenominal adjectives. Semantically, this is borne out by the data. Consider the NP in (93) below. It describes an entity as being in fact Chinese. That is, the piece of semantics for that NP will be equivalent to $\lambda P.|((D_e - D) \cap C) \cap P| > 0$ (where D_e is the model's domain, D the set of doctors and C the set of Chinese entities, giving the adjective "falso" the semantics $\lambda Q.D_e - Q$), with no mismatch between syntactic and semantic scope. It will not be $\lambda P.|(D_e - (D \cap C)) \cap P| > 0$, the semantics for the example in (80) ("um falso médico chinês" — *a false Chinese doctor*).

(93) um falso médico que é chinês
a fake doctor who is Chinese
a fake doctor that is Chinese

Semantically, restrictive relative clauses are under the scope of cardinals. Consider the sentence in (94). If M is the set of movies, BM is the set of bad movies and S is the set of entities that "I saw there", the meaning of (94) is $M \cap S \subseteq BM \wedge |M \cap S| = 3$, not $M \cap S \subseteq BM \wedge |M| = 3$.

(94) Todos os exactamente três filmes que lá vi eram maus.
all the exactly three movies that there I saw were bad
All exactly three movies I saw there were bad.

Under the assumption that, if there is no reason to assume the contrary, syntactic scope matches semantic scope, we therefore want relative clauses to attach lower than cardinals. Similar data can be envisaged for ordinals, but we will not present them for the sake of brevity. Semantic considerations cannot help us determine the relative scope between relative clauses and markers of indefinite specifics, because the latter contribute no relations to the resulting semantics. Since cardinals, ordinals and markers of indefinite specifics all occupy the same NP slot, we assume that relative clauses attach lower than all these elements. To force this attachment, we can simply add the following constraints to the type *rel-marking*:

$$\begin{bmatrix} \textit{rel-marking} & \\ \textsc{mk-val} & \begin{bmatrix} \textsc{cardinal} & \textit{absent} \\ \textsc{ordinal} & \textit{absent} \\ \textsc{indef-spec} & \textit{absent} \\ \textsc{tal} & \textit{absent} \\ \textsc{possessive} & \textit{absent-or-posthead-present} \\ \textsc{qualquer} & \textit{absent-or-posthead-present} \\ \textsc{outro} & \textit{absent-or-posthead-present} \end{bmatrix} \end{bmatrix}$$

Recall that the feature MK-VAL is unified between MARKER|MARK and MARKER|SELECT|LOCAL|CAT|MARKI under the HEAD attribute of relative clauses, so the constraints just presented on *rel-marking* effectively make relative clauses select for constituents that lack all of these elements.

The constraint on the feature POSSESSIVE also makes prenominal possessives outscope relative clauses, since prenominal possessives precede cardinals. The constraint on the feature TAL is because the element "tal" (*such*) must also precede cardinals.

It is worth mentioning that there are data that this analysis does not contemplate:

(95) a sugestão$_N$ [$_{RelCl}$ que foi mencionada] [$_{COMP}$ de que seria impossível
the suggestion that was mentioned that it would be impossible
proceder de outro modo]
to act in a different way
the suggestion that was mentioned that it would be impossible to act in a different way

In (95) there is a relative clause, bracketed with RelCl, intervening between the head noun and its sentential complement, bracketed with COMP, for complement. However, according to our system of constraints, relative clauses must attach to projections with already saturated complements. This is because Head-Complement constructions constrain the head daughter to have MARKING of type *basic-marking*, as presented in Section 8.6.8. This constraint was necessary in order to force PP complements to precede relative clauses, as explained in that section.

We believe that this sort of situation only arises in specific cases (sentential complements) and that they should receive a special treatment, which we do not develop here.

8.6.11 Postnominal Demonstratives and Postnominal Possessives

In Position VIII we can find other elements besides adverbial PPs, AdvPs and APs and complements, which have been covered. These other elements are postnominal demonstratives, possessives and universal quantifiers:

(96) a. A bicicleta essa está estragada.
the bicycle that is broken
That bicycle is broken.

b. Chegaram várias cartas tuas.
arrived several letters yours
There arrived several letters of yours.

c. Desapareceram as cartas todas.
disappeared the letters all
All the letters disappeared.

We will not address postnominal universal quantifiers. They motivate a more complicated composition of semantics, because they introduce a quantifier relation with scope over the rest of the semantic material of the NP they are in and yet occur in a position (Position VIII) that does not have syntactic scope over that material. Consider the following example, bracketed according to the syntactic structure that is assumed:

(97) Desapareceram [as [[cartas todas] da Ana.]]
disappeared the letters all of the Ana
All of Ana's letters disappeared.

We leave this issue to future work.

8.6.11.1 Postnominal Demonstratives

Postnominal demonstratives are possible in some dialects of Portuguese. They are confined to NPs introduced by a definite article. They do not co-occur with prenominal demonstratives and do not iterate:

(98) a. A bicicleta essa está estragada.
the bicycle that is broken
That bicycle is broken.

b. * Uma bicicleta essa está estragada.
a bicycle that is broken

c. * Essa bicicleta essa está estragada.
that bicycle that is broken

d. * Esta bicicleta essa está estragada.
this bicycle that is broken

e. * A bicicleta essa essa está estragada.
the bicycle that that is broken

The feature DEMONSTRATIVE under MK-VAL is used to block the co-occurrence of prenominal and postnominal demonstratives and to prevent demonstratives from iterating. This feature is used as it can be expected from the use of the other features of MK-VAL as presented before: prenominal and postnominal demonstratives select for sisters with MARKING|MK-VAL|DEMONSTRATIVE of type *absent* and have MARK|MK-VAL|DEMONSTRATIVE with the value *present*, the remaining functors unify the DEMONSTRATIVE attributes under the paths SELECT|LOCAL|CAT|MARKING|MK-VAL and MARK|MK-VAL. In order to block the co-occurrence of indefinite determiners with postnominal demonstratives, indefinite determiners also select for sisters with an *absent* DEMONSTRATIVE.

Prenominal demonstratives and postnominal demonstratives must come in the lexicon in different entries, or related by lexical rules, because the prenominal ones are determiners and carry quantifier semantics. The constraints on MARKING and MARK are also different between these two sets of items. Prenominal demonstratives have a HEAD of type *determiner*, while postnominal ones must have *marking* constraints almost identical to the other elements in the same slot (postnominal adjectives, etc.). The word order between functor and head is also different. The head of postnominal demonstratives looks like:

$$\begin{bmatrix} \textit{postnominal-demonstrative} \\ \text{MARKER} \begin{bmatrix} \textit{post-only-marker} \\ \text{MARK} \begin{bmatrix} \textit{n-marking} \\ \text{MK-VAL} \begin{bmatrix} \text{DEMONSTRATIVE} & \textit{present} \\ \text{POSSESSIVE} & \boxed{1} \\ \text{QUALQUER} & \boxed{2} \\ \text{TAL} & \boxed{3} \\ \text{OUTRO} & \boxed{4} \\ \text{CARDINAL} & \boxed{5} \\ \text{ORDINAL} & \boxed{6} \\ \text{INDEF-SPEC} & \boxed{7} \end{bmatrix} \end{bmatrix} \\ \text{SELECT}|\text{LOCAL}|\text{CAT} \begin{bmatrix} \text{HEAD } \textit{noun} \\ \text{MARKING} \begin{bmatrix} \textit{basic-marking} \\ \text{MK-VAL} \begin{bmatrix} \text{DEMONSTRATIVE} & \textit{absent} \\ \text{POSSESSIVE} & \boxed{1} \\ \text{QUALQUER} & \boxed{2} \\ \text{TAL} & \boxed{3} \\ \text{OUTRO} & \boxed{4} \\ \text{CARDINAL} & \boxed{5} \\ \text{ORDINAL} & \boxed{6} \\ \text{INDEF-SPEC} & \boxed{7} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

To make the composition of semantics easier with demonstratives, we view determiner demonstratives as carrying two semantic relations: a quantifier relation and an intersective relation in the restrictor of the quantifier. When a demonstrative is used deictically, the second relation can be semantically considered to be roughly similar to the relation of adverbs like *here* or *there*. In this case a noun phrase like *that car* is considered semantically close to a noun phrase like *the car there*, and *this car* to *the car here*. There is some empirical support to this analysis, as the demonstrative and the adverb must agree with respect to deixis:

(99) a. Esta bicicleta aqui está estragada.
this bicycle here is broken
This bicycle here is broken. (the bicycle near me/us)

b. Essa bicicleta aí está estragada.
that bicycle there is broken
That bicycle there is broken. (the bicycle near you)

c. Aquela bicicleta ali está estragada.
that bicycle there is broken
That bicycle there is broken. (the bicycle away from me and you)

d. * Esta bicicleta aí está estragada.

e. * Esta bicicleta ali está estragada.

f. * Essa bicicleta aqui está estragada.

g. * Essa bicicleta ali está estragada.

h. * Aquela bicicleta aqui está estragada.

i. * Aquela bicicleta aí está estragada.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h6}\ h \\ \text{BODY} & \boxed{h5}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_esse\_a\_rel} & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{e8}\ e \\ \text{ARG1} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_carro\_n\_rel} & \\ \text{LBL} & \boxed{h7} \\ \text{ARG0} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_avariar\_v\_rel} & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h6} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 8.19: MRS of a sentence with a demonstrative. The corresponding sentences are "esse carro avariou" and "o carro esse avariou" (*that car broke down*).

The names of these predicates in the restrictor of the quantifier are the lemma of the demonstrative — we do not commit to saying that they are identical to adverbial relations. They are obviously not so when demonstratives are employed anaphorically, in which case these predicates are assumed to take a different meaning. We cannot detect automatically with the grammar whether a demonstrative is being used anaphorically or deictically, so the relations visible in the MRSs are meant to be underspecifications.

Postnominal demonstratives introduce a single, plain intersective, relation in the MRS, and the quantifier relation comes from the definite article. Prenominal demonstratives introduce two relations in the MRS representation, a similar intersective one and a quantifier relation. We give them semantics similar to that of a postnominal demonstrative co-occurring with a definite article. That is, the quantifier relation of prenominal demonstratives is the same as that of definite articles, thus treating the following examples as paraphrases of one another:

(100) a. o carro esse
the car that
that car

b. esse carro
that car
that car

MRSs for these two cases are shown in Figure 8.19. The LBL of the *_esse_a_rel* relation is the LTOP of the demonstrative determiner's sister: demonstrative determiners unify this LBL with the path SELECT|LOCAL|CONT|HOOK|LTOP under their HEAD.

The attribution of two relations to prenominal demonstratives is also useful in the context of a predeterminer, as in the following example:

(101) [$_{\text{NP}}$ Todos esses carros] avariaram.
all those cars broke down
All those cars broke down.

In Section 8.6.2 it was stated that determiners co-occurring with predeterminers contribute no quantifier semantics. The fact that demonstratives introduce two relations in the MRS gives us

$$\begin{bmatrix} \textit{mrs} & & \\ \text{LTOP} & \boxed{h1}\ h & \\ \text{INDEX} & \boxed{e2}\ e & \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{\_todo\_q\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h6}\ h \\ \text{BODY} & \boxed{h5}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_esse\_a\_rel} & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{e8}\ e \\ \text{ARG1} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_carro\_n\_rel} & \\ \text{LBL} & \boxed{h7} \\ \text{ARG0} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{\_avariar\_v\_rel} & \\ \text{LBL} & \boxed{h9} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \right\rangle & \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h6} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle & \end{bmatrix}$$

Figure 8.20: MRS of a sentence with predeterminer and a demonstrative. The sentence is "todos esses carros avariaram" (*all those cars broke down*).

a simple way to distinguish between semantic representations of sentences with a predeterminer and a demonstrative determiner from semantic representations of sentences with a predeterminer and a definite article. More specifically, there have to be versions of prenominal demonstratives that do not introduce quantifier semantics as well, with *marking* constraints similar to the ones on the vacuous definite articles presented in Section 8.6.2. These determiners are however not semantically vacuous, they still introduce the special predicate in the restrictor of the quantifier. The quantifier relation is introduced by the predeterminer, as before.

In the presence of a predeterminer, NPs with postnominal demonstratives and NPs with prenominal demonstratives have similar MRSs, too. An example MRS is in Figure 8.20. It is the MRS for the sentences "todos esses carros avariaram" and "todos os carros esses avariaram" (*all those cars broke down*).

To control the co-occurrence restrictions between the set of determiners and postnominal demonstratives in (98), the relevant determiners are constrained to select for a sister with DEMONSTRATIVE of type *absent*.

8.6.11.2 Postnominal Possessives

Postnominal possessives are constrained to occur in indefinite NPs or NPs introduced by a demonstrative (Section 8.6.3).

We use the feature POSSESSIVE under MK-VAL to control the distribution of possessives.

The HEAD of possessives has the additional constraints under the POSTHEAD feature (in addition to the constraints presented for this type in Section 8.6.3):

$$\begin{bmatrix} \textit{possessive} & \\ \text{MARKER}|\text{POSHEAD} & \begin{bmatrix} \text{SELECT}|\text{LOCAL}|\text{CAT}|\text{MARKING}\ \textit{n-marking} & \\ \text{MARK} & \begin{bmatrix} \textit{n-marking} & \\ \text{MK-VAL}|\text{POSSESSIVE} & \textit{posthead-present} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The co-occurrence of prenominal and postnominal possessives is prevented because they all select for a sister with POSSESSIVE *absent*, they all have MARK|MK-VAL|POSSESSIVE with the value *present*.

The same attributes (the ones under MK-VAL) are also used to constrain the different distribution of prenominal and postnominal possessives, since the types appropriate for these features include information relating to realization and also word order.

Possessives have *posthead-present* under HEAD|MARKER|POSTHEADIMARK|MK-VAL|POSSESSIVE and the value *prehead-present* under HEAD|MARKER|PREHEADIMARK|MK-VAL|POSSESSIVE. Definite articles, which cannot co-occur with postnominal possessives, select for sisters with SYNSEM|LOCAL|CAT|MARKING|MK-VAL|POSSESSIVE *absent-or-prehead-present*. As usual, all functors that are not possessives must percolate this feature by unifying their MARKER|MARK|MK-VAL|POSSESSIVE with their MARKER|SELECT|LOCAL|CAT|MARKING|MK-VAL|POSSESSIVE.

Indefinite determiners select a sister with MARKING|MK-VAL|POSSESSIVE of type *absent-or-posthead-present*, since they cannot co-occur with prenominal possessives. This is also true of the cardinal determiners presented in Section 8.6.5.

Prenominal demonstratives, which can co-occur with both prenominal and postnominal possessives, do not constrain the feature POSSESSIVE.

8.6.12 Gerunds

Gerund forms that occur as modifiers are derived from the lexical entries for verbs via two inflectional rules.

One of these rules produces gerund forms that must modify a noun-headed constituent, the other rule produces gerund forms that must modify a verb-headed constituent.

In the implementation of LXGram, all modifying gerunds are given intersective semantics.

For gerunds that modify nominal constituents, the ARG1 of the gerund's relation (the argument corresponding to the subject of that verb) is the index of the head noun, accessible in the INDEX attribute of the modified element.

The semantics for the example NP below ("um livro descrevendo a Irlanda", *a book describing Ireland*) is thus similar to $\lambda P.\ proper\_q(x,\ named(x,\ "Irlanda"),\ \_um\_q(y,\ \_livro\_n(y)\ \wedge\ \_descrever\_v(y,x),\ \ P(y)))$: this NP is given semantics similar to "um livro que descreve a Irlanda" (*a book that describes Ireland*).

(102) um livro descrevendo a Irlanda
a book describing the Ireland
a book describing Ireland

For gerunds that modify verb phrases, the ARG1 feature of the gerund's relation is identified with the index of the modified verb's subject (the subject of the gerund is the subject of the main verb). This value is visible in the XARG feature of the modified constituent. The following example thus receives semantics similar to $\_pronoun\_q(x, pronoun\_n(x), \_sair\_v(e_1, x) \wedge \_correr(e_2, x))$.

(103) a. Saíram correndo.
they left running
They left running.

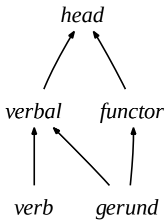
This analysis does not contemplate expressions like "considerando que..." (*considering that...*), where the subject of the gerund form may be independent of the main verb's subject.

Because the value of the ARG1 feature of the gerund's relation is obtained from different attributes of the modified constituent (XARG if it is a verb phrase, INDEX if it is nominal), two separate rules are used for gerunds modifying nouns and verbs.

For gerunds modifying verb phrases, we add an additional relation, called *gerund_rel*, in order to make it explicit which relation corresponds to the gerund. For our previous example, "saíram correndo" (*they left running*), LXGram produces a semantic representation similar to

$\_pronoun\_q(x, pronoun\_n(x), \_sair\_v(e_1, x) \wedge gerund(e_2, e_1) \wedge \_correr\_v(e_2, x))$. The fact that e_2 (the first argument of the *gerund_rel* relation) is the event variable of the *_correr_v_rel* relation indicates that this is the relation corresponding to the gerund.

The HEAD type of gerunds has to be different from the HEAD type of all other verbal forms: gerunds are modifiers (they must have the MARKER feature under head, and therefore must inherit from the type *functor*), whereas non gerund verb forms cannot head modifiers (and so they must not inherit from *functor*). The following simplified type hierarchy presents the relevant types employed in LXGram to control this:



The lexical entries for verbs come with the value *verbal* for the HEAD attribute. This type is specialized as either *gerund* or *verb* in all inflectional rules. The type *gerund* is used in the rules that produce gerund forms, and contains the constraints that control attachment of phrases headed by a gerund form. The type *verb* is used in all other morphological rules. It does not inherit from *functor*, making it impossible for non-gerund verb forms to occur as the functor daughter of head-functor constructions.

The constraints on the features MARKING and MARK of gerunds are similar to the constraints for prepositions and adverbs.

This implementation of gerunds is experimental, and a final solution will be sought in the next development phases.

8.6.13 Other NP Elements with a Special Treatment

LXGram gives a special treatment to other NP elements, namely the elements "outro" (*other*), "qualquer" (*any*) and "tal" (*such*). These elements have a special syntax. They do not fit cleanly in the NP positions identified in Section 8.6. None of these elements can occur multiple times in an NP; therefore there is an attribute under MK-VAL for each one of them, recording their presence (the features OUTRO, QUALQUER and TAL).

The element "outro" can appear in NP initial position, but also in positions that are internal to the NP. In the latter case, its distribution is different from that of adjectives, because it can precede cardinals, for instance. The following sentences illustrate this point.

(104) a. outra bicicleta
another bicycle
another bicycle

b. a outra bicicleta
the other bicycle
the other bicycle

c. as duas outras bicicletsa
the two other bicycles
the two other bicycles

d. as outras duas bicicletas
the other two bicicles

the other two bicicles

The order between “outro” and prenominal possessives is also free:

(105) a. a minha outra bicicleta
the my other bicycle
my other bicycle

b. a outra minha bicicleta
the other my bicycle
my other bicycle

The element “tal” also exhibits special behavior:

(106) a. a tal bicicleta
the such bicycle
that bicycle

b. essa tal bicicleta
that such bicycle
that one bicycle

c. a minha tal bicicleta
the my such bicycle
that bicycle of mine

d. a tal minha bicleta
the such my bicycle
that bicycle of mine

e. a tal bicicleta minha
the such bicycle of mine
that bicycle of mine

f. tal bicicleta minha
such bicycle of mine
that bicycle of mine

g. * tal minha bicicleta
such my bicycle

h. tais duas bicicletas
such two bicycles
two such bicycles

i. duas tais bicicletas
two such bicycles
two such bicycles

j. as tais duas bicicletas
the such two bicycles
those two bicycles

k. as duas tais bicicletas
the two such bicycles
those two bicycles

l. * a tal tal bicicleta
the such such bicycle

The item "qualquer" is also idyossincratic:

(107) a. qualquer bicicleta minha
any bicycle of mine
any bicycle of mine

b. * qualquer qualquer bicicleta minha
any any bicycle of mine

c. quaisquer duas bicicletas
any two bicycles
any two bicycles

d. duas quaisquer bicicletas
two any bicycles
any two bicycles

e. duas bicicletas quaisquer
two bicycles any
any two bicycles

f. * quaisquer bicycletas quaisquer
any bicycles any

g. * as quaisquer bicicletas
the any bicycles

h. * as bicicletas quaisquer
the bicycles any

These features OUTRO, TAL and QUALQUER are used like the other features under MK-VAL: they are unified between MARKING and MARK in the other functors, and constrained with the values *present* or *absent* in these elements. They are used in a way analoguous to the way the other features are used, as presented before, by placing constraints on these feature in the various functors.

Some of these constraints have been presented before. For instance, markers of indefinite specific NPs select for a sister node with the value *absent* for the feature TAL (see Section 8.6.4), because of ungrammatical examples like the following one:

(108) * certas tais bicicletas
certain such bicycles

The item "tal" also selects a sister with INDEF-SPEC of type *absent*, in order to block the following example:

(109) * tais certas bicicletas
such certain bicycles

8.6.14 Postponed Coverage or Known Limitations

Partitives are not contemplated by the current implementation.

LXGram contains no implementation yet of noun-noun compounds, like "palavra-chave" (*keyword*), "governo fantoche" (lit. *puppet government*), "experiência piloto" (*pilot experiment*).

8.7 Noun Ellipsis

In some NPs there is no overt head noun. Some examples are in (110).

(110) a. a casa azul e [a - verde]
the house blue and the green
the blue house and the green one

b. algumas crianças com chapéu e [algumas - com boné]
some children with hat and some with cap
some children in hats and some in caps

c. Os que podem ajudar nunca ajudam.
the who can help never help
The people who can help never do so.

The examples in (110a) and (110b) are however very different from the one in (110c) with respect to the semantics of the missing nouns. In (110a) the noun form "casa" (*house*) is recoverable from context, but in (110c) the missing noun (something close to *people*) is independent of context and its semantics can be described as generic.

For this reason, the phenomenon in (110c) has been referred to as *people* deletion [Pullum, 1975]) or null-N generics [Nerbonne and Mullen, 2000]. Here we will adopt the designation *missing-N generics*, in order to remain neutral to the status of non-realized elements.

On the other hand, the phenomenon in (110a) is known as noun ellipsis.

In LXGram these constructions are handled via unary rules that produce a mother node with a HEAD feature with the type *noun*. The daughter of these rules is any element that selects for a noun headed constituent via their SELECT feature. Since we cannot distinguish between missing noun generics and noun ellipsis on the basis of syntax, this construction adds a relation to the MRS with the name *ellipsis-or-generic_n_rel*.

Implementation details are provided in the next sections.

8.7.1 ARGS, HEAD-DTR and NON-HEAD-DTR

The order of the daughters of phrasal constituents is denoted in the LKB by the order of elements in the list-valued feature ARGS. Furthermore, attributes like HEAD-DTR and NON-HEAD-DTR are merely pointers to these elements, useful when one wants to abstract from word order.

Nothing requires that these daughter features point to an existing element of ARGS, though. That is, it is possible to have constructions with the two features, HEAD-DTR and NON-HEAD-DTR, but with an ARGS list of less than two elements.

This is a way to model a class of missing syntactic constituents. Assuming that phrases are binary at most (this is enforced in LXGram and several other computational HPSGs), these constructions are prototypically unary, but have semantic or syntactic properties of some other binary constructions.

The difference between ARGS and daughter features (HEAD-DTR and NON-HEAD-DTR) has no theoretical status in HPSG, and the attribute ARGS is specific to the LKB. But we can make a conceptual distinction between them, and give them a theoretical status. The feature ARGS denotes the realized daughters of a phrase, whereas the daughter features (like HEAD-DTR and NON-HEAD-DTR) include them as well as elements that correspond to empty constituents.

There are thus two dimensions: the HEAD-DTR and NON-HEAD-DTR level, which abstracts from the possibility of non-realized constituents, and the ARGS level, which is more superficial in this respect. ARGS is also the best place where the Principle of Canonicality can be enforced (all elements of ARGS are required to have SYNSEMs of type *canonical-synsem*). This is what is done in LXGram (see Section 8.6.1).

The implementation of the missing noun phrases is an interesting case to justify these two levels. It is explained in Section 8.7.2.

8.7.2 Implementation

The phrase types for missing noun constructions are descendants of *basic-missing-noun-phrase* where both the HEAD-DTR and the NON-HEAD-DTR features are present but ARGS has a single element in it, corresponding to the non-head daughter. The type *basic-missing-noun-phrase* is a subtype of *basic-head-functor-phrase*, where several constraints for the Head-Functor constructions are stated. These constraints are inherited by *basic-missing-noun-phrase.* They have been presented in Section 5.3.

In the type *basic-missing-noun-phrase* it must be specified which daughter is realized. We chose to do it in a supertype, *head-missing*, assuming that there can be other constructions with a singleton ARGS but with both HEAD-DTR and NON-HEAD-DTR features, which could be defined to also inherit from *head-missing*:

$$\begin{bmatrix} \textit{head-missing} \\ \text{HEAD-DTR}|\text{SYNSEM } \textit{non-canonical-synsem} \\ \text{NON-HEAD-DTR } \boxed{1} \\ \text{ARGS } \left\langle \boxed{1} \right\rangle \end{bmatrix}$$

The attribute SYNSEM of the missing daughter is constrained to be of the type *non-canonical-synsem*, in view of the fact that it is not realized.

Its counterpart type, *non-head-missing*, is also part of the hierarchy of phrse types and is specified to have the expected constraints, namely:

$$\begin{bmatrix} \textit{non-head-missing} \\ \text{HEAD-DTR } \boxed{1} \\ \text{NON-HEAD-DTR}|\text{SYNSEM } \textit{non-canonical-synsem} \\ \text{ARGS } \left\langle \boxed{1} \right\rangle \end{bmatrix}$$

The interesting part of this design is that, in order to add noun semantics and constrain the type of the HEAD and MARKING features (to be *noun* and *n-marking* respectively) that the functor feeding the *basic-missing-noun-phrase* will see under its SELECT feature, one can put this information under the HEAD-DTR attribute. The basic machinery put in place to percolate syntactic information from the daughters in headed phrases and Head-Functor schemata fills the appropriate values in the mother node — it is completely inherited from supertypes. All that is required is that the supertypes never constrain ARGS, and use HEAD-DTR and NON-HEAD-DTR instead.

This approach is taken even further. Since the constraints on the HEAD-DTR feature effectively consist of the definition of a noun, HEAD-DTR can simply be constrained to be of a type that is a supertype of lexical items for nouns.

In LXGram there is a type, *noun-common-0comps-3p-sign*, that is the supertype of all lexical types for nouns with no complements that do not accept second-person readings (see Section 8.3). It includes constraints that determine the SYNSEM|LOCAL|CAT|HEAD feature to be of type *common-noun* (see Section 8.1) and the attribute SYNSEM|LOCAL|CAT|MARKING must have a value that denotes that this element is not a full NP. Under SYNSEM|LOCAL|CONT, HCONS is empty and RELS includes a single relation with an ARG0 of type *ref-index* (the real type name of the variables that show up in MRSs with type x) structure-shared with HOOK|SARG (see Section 5.11) and HOOK|LTOP is unified with the LBL of that relation:

$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{noun-common-0comps-3p-sign}} \\
\text{SYNSEM}|\text{LOCAL} & \left[\begin{array}{ll}
\text{CAT} & \left[\begin{array}{ll}
\text{HEAD} & \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{common-noun}} \\ \text{AGR}|\text{PERSON} & \textit{3rd} \end{array}\right] \\
\text{VAL}|\text{COMPS} & \langle\rangle \\
\text{MARKING} & \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{non-saturated-marking}} \\
\text{DEMONSTRATIVE} & \textit{absent} \\
\text{POSSESSIVE} & \textit{absent} \\
\text{TAL} & \textit{absent} \\
\text{QUALQUER} & \textit{absent} \\
\text{OUTRO} & \textit{absent} \\
\text{CARDINAL} & \textit{absent} \\
\text{ORDINAL} & \textit{absent} \\
\text{INDEF-SPEC} & \textit{absent} \end{array}\right]
\end{array}\right] \\
\text{CONT} & \left[\begin{array}{ll}
\text{HOOK} & \left[\begin{array}{ll} \text{LTOP} & \boxed{1} \\ \text{SARG} & \boxed{2} \end{array}\right] \\
\text{RELS} & \left\{\left[\begin{array}{lll} \text{LBL} & \boxed{1} & \textit{handle} \\ \text{ARG0} & \boxed{2} & \textit{ref-index} \end{array}\right]\right\} \\
\text{HCONS} & \{\}
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

The value of the feature MARKING in this type is explained in Section 8.7.3. The lexical types that inherit from *noun-common-0comps-3p-sign* and correspond to overt nouns further constrain this feature with the value *n-marking*, as presented above.

A descendant of *noun-sign* is *covert-noun-sign*, representing a noun that has no phonetic realization. The semantics specific to missing noun constructions (LXGram does not resolve the antecedent of noun ellipsis) is specified here:

$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{covert-noun-sign}} \\
\text{SYNSEM} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{unexpressed-synsem}} \\
\text{LOCAL} & \left[\text{CONT}|\text{RELS}\left\{\left[\text{PRED}\ \textit{ellipsis-or-generic\_n\_1\_rel}\right]\right\}\right]
\end{array}\right]
\end{array}\right]$$

The constraint on the type of its SYNSEM attribute (type *unexpressed-synsem*) denotes the fact that this noun is not realized.

This lexical type is not used in lexical entries, since our analysis does not resort to null constituents, but it can be used in the definition of the constructions with missing nouns. These constructions specify their head daughter to be a *covert-noun-sign*:

$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{basic-missing-noun-phrase}} \\
\text{HEAD-DTR} & \textit{covert-noun-sign}
\end{array}\right]$$

All the properties specific to *basic-missing-noun-phrase* follow immediately from the constraints inherited from its supertypes and the constraints on *covert-noun-sign*, and do not have to be stated as specific constraints in the *basic-missing-noun-phrase* type.

In particular, the daughter of this construction must be an element that selects for a noun headed constituent, since the constraints inherited from *basic-head-functor-phrase* unify the SELECT feature of the non-head daughter with the SYNSEM under HEAD-DTR.

The MARK feature of the non-head daughter is also unified with the MARKING of the mother node via the inherited constraints. Therefore, if a determiner feeds this rule, a full NP is produced, but if e.g. an adjective is the daughter then the resulting node must combine with another element before forming a full NP.

The HEAD feature of the mother node is also constrained to be of the type *common-noun*, since it is unified with the HEAD feature of the HEAD-DTR in supertypes.

The advantages of this implementation are:

- No *ad hoc* constraints on missing noun phrases are needed to add noun semantics or to constrain the value of the HEAD feature of the mother node or of the SELECT feature of the functor daughter. The constraints necessary to compose semantics are also inherited from very general supertypes.
- The constraints common to overt nouns and the missing head are stated in a single place. Furthermore, these constraints basically define what a noun is. This makes it easier to change the implementation. For instance, changes in the type hierarchy of *marking* that require changes in the value of MARKING of nouns do not require changes both in the lexical types of nouns and in the definition of missing noun phrases.
- The constraints that define what a noun is are encapsulated in the type used to constrain the HEAD-DTR feature and not directly stated in the type for missing noun phrases.

The main disadvantage is that the feature structures for missing noun phrases will be substantially larger, since the feature structure for an entire lexical item will be present under HEAD-DTR. Note however that it does not imply more unification operations at run time, since no node will be unified with the entire HEAD-DTR attribute, as it is not an element of ARGS.

8.7.3 Predeterminers in Missing Noun Constructions

In LXGram all nouns come in the lexicon with the value *n-marking* for the feature MARKING. However, in the syntax rules for missing nouns, the daughter of those rules is constrained to be a functor selecting for a noun-headed constituent with MARKING of type *non-saturated-marking* (as presented above), rather than *n-marking.* The reason for this mismatch is the peculiar behavior of the predeterminer “todos” (*all*). This element can appear in noun ellipsis (111a) or missing-N generic constructions (111b).

(111) a. O João comprou maçãs e todas estavam podres.
the João bought apples and all were rotten
João bought apples and all (of them) were rotten.

b. Todos são livres.
all are free
All (people) are free.

When this element appears in NPs with an overt head, a determiner must also be present in the case of European Portuguese, as reported in Section 8.6.2. The analysis presented in that section resorted to two lexical entries for this item, one of them common to European and Brazilian Portuguese, where the presence of a determiner is required, and another specific to Brazilian Portuguese, where it is not.

For the first entry, a lexical specification was employed in predeterminers according to which they select for a nominal projection with a value of MARKING subsumed by *non-saturated-det-marking*. This constraint makes it incompatible with constituents that bear the value *n-marking* for this attribute, but it is compatible with the value *non-saturated-marking*. This is the reason why the value of MARKING of missing nouns is more abstract than that of overt nouns.

The second entry, specific to Brazilian Portuguese, is also allowed in the missing noun constructions, but sentences like the ones in (111) are possible in European Portuguese, too. Therefore, we would like the item exclusive to Brazilian Portuguese to be blocked, since that would just multiply parses. The item exclusive to Brazilian Portuguese is constrained to select for a constituent with a synsem of type *canonical-synsem*. This prevents it from feeding the rules for missing nouns, since the HEAD-DTR of these constructions has a SYNSEM of type *unexpressed-synsem* (see the constraints on *covert-noun-sign* above), that is incompatible with *canonical-synsem*.

8.8 Bare NPs

The implementation of bare NPs (NPs lacking a determiner) is very similar to the implementation of constructions with missing nouns. Some examples of bare NPs are given below.

(112) a. Desapareceram livros da biblioteca.
disappeared books from the library
There disappeared books from the library.

b. Compraram novos livros para a biblioteca.
they bought new books for the library
They bought new books for the library.

c. ??/* Livros desapareceram da biblioteca.
books disappeared from the library

As the last example shows, bare NPs are not generally acceptable as preverbal subjects. Currently, this possibility is not blocked by LXGram.

In LXGram, bare NPs are implemented as inheriting from a type for Head-Functor constructions and also from *non-head-missing* (see the definition of this type in Section 8.7.2). The missing functor is constrained to be a determiner, in a way similar to the way that the head daughter of missing noun constructions is constrained to be a noun. The head daughter of bare-NP constructions is therefore automatically constrained to be headed by a noun and bear an appropriate value for the feature MARKING. Also, the semantics of a quantifier relation that this rule must introduce comes from the constraints on the missing daughter. There are additional constraints on the features under MK-VAL of the head daughter, which have been stated in Section 8.6.5.

8.9 Postponed Coverage or Known Limitations

The implementation of appositive modification is postponed to a phase after punctuation has been implemented more substantially, since appositive modifiers are generally written between commas.

Chapter 9

Verb Phrases

9.1 Implemented Subcategorization Frames

The following subsections describe the implementation of several subcategorization frames.

9.1.1 Copular verbs

Copular verbs are analysed as raising the subject of their complement. Every element that can be a predicate selected by a copula must thus have a non empty SUBJ list. This allows these elements to constrain the type of subject of which they can be a predicate.

An example follows. Sentences (113) indicate that the CP that co-occurs with an adjective of the kind of "óbvio" (*obvious*) must be its subject, as it cannot co-occur with another subject. A color adjective, however, cannot have a sentential subject (114). Adjectives can thus control the type of subjects they take in their SUBJ list.

(113) a. Isso é óbvio.
that is obvious
That is obvious.

b. É óbvio que isso é assim.
is obvious that that is so
It is obvious that that is like that.

c. * Isso é óbvio que isso é assim.
that is obvious that that is so

(114) * É verde que isso é assim.
is green that that is so

9.1.2 Intransitive verbs

Intransitive verbs are implemented as carrying a unary relation and selecting for an NP subject and no complement. The subject realizes the single argument of the verbal relation. In the lexical entry for these verbs, the INDEX of the subject is thus unified with the ARG1 of the verbal relation.

9.1.3 Transitive verbs

Direct transitive verbs are implemented in LXGram as selecting for an NP subject and an NP complement. They introduce a binary relation in the semantic representation. The first argument of that relation is realized by the subject, and the second argument by the complement. Therefore, the ARG1 feature of the verb's relation is unified with the INDEX of the subject, and the ARG2 feature of that relation is unified with the INDEX of the NP complement.

The CASE feature of direct transitive verbs is constrained to be of the type *accusative*.

Indirect transitive verbs are also implemented. The semantic constraints on these verbs are just like the ones for direct transitive verbs. Syntactically, they select for a PP complement headed by the preposition "a".

There is a lexical rule in LXGram that indirect transitive verbs can optionally undergo. This is the *dative-pp-1st_comp-alternation* lexical rule. This rule changes the PP complement into an NP complement with dative case. Only clitic pronouns can have dative case in LXGram, since common and proper nouns have the value *non-dative* for their feature CASE, a type incompatible with the type *dative*.

9.1.4 Ditransitive verbs

In the implementation, ditransitive verbs select for an NP subject and two complements. The first complement is an accusative NP, and the second one is a PP similar to the complement of indirect transitive verbs.

These verbs contribute a three-place relation to the MRS representation. The first argument is realized by the subject, the second argument by the first complement, and the third argument is realized by the PP complement.

There is also a lexical rule to turn the second complement of these verbs, the PP complement, into a dative NP. It's the *dative-pp-2nd$_c$omp-alternation* lexical rule, and it is similar to the *dative-pp-1st_comp-alternation* described above.

9.1.5 Verbs with an inherent clitic complement

In LXGram, the type *verb-inherent_clitic-lex* implements the subcategorization frame of verbs that take an NP subject and a single complement that has to be an inherent clitic.

An example of such a verb is the verb "constipar-se" (*to catch a cold*). Figure 9.1 shows the MRS for a sentence with this verb. Verbs like this one have two syntactic dependents, but only one argument for their relation, corresponding to the subject.

$$\left[\begin{array}{ll}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle
\left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_o\_q\_rel}\langle 0{:}2\rangle} \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h\end{array}\right],
\left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_gato\_n\_rel}\langle 3{:}8\rangle} \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4}\end{array}\right],
\left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_também\_a\_rel}\langle 9{:}15\rangle} \\ \text{LBL} & \boxed{h8}\ h \\ \text{ARG0} & \boxed{e9}\ e \\ \text{ARG1} & \boxed{h10}\ h\end{array}\right],
\left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_constipar-se\_v\_rel}\langle 19{:}28\rangle} \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4}\end{array}\right]
\right\rangle \\
\text{HCONS} & \left\langle
\left[\begin{array}{ll}\textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h8}\end{array}\right],
\left[\begin{array}{ll}\textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7}\end{array}\right],
\left[\begin{array}{ll}\textit{qeq} & \\ \text{HARG} & \boxed{h10} \\ \text{LARG} & \boxed{h11}\end{array}\right]
\right\rangle
\end{array}\right]$$

Figure 9.1: MRS for the sentence "Os gatos também se constipam" (*Cats catch colds, too*).

Inherent clitic pronouns are implemented as personal pronouns clitics (with respect to their syntactic properties) that carry no semantics. They have empty RELS and HCONS features.

The value of the feature SYNSEM|LOCAL|CONT|HOOK|INDEX is *expl-ind* for these items. The type *expl-ind* is used for expletives (inherent clitics and expletive nominative pronouns) and is incompatible with the type for the features ARG0, ARG1, ARG2, etc. of relations. This is the way that dummy pronouns are prevented from realizing complements of verbs like transitive verbs, whose complement realizes an argument of the corresponding verbal relation.

Verbs that have an inherent clitic complement simply select for a complement with a HEAD of type *noun* and an INDEX attribute of the type *expl-ind*. The complement of these verbs can only be an inherent clitic, since the type for this feature INDEX of semantically contentful NPs is the incompatible type *ref-ind* (a referential index). Nominative expletive pronouns (like "ele", *it*, that can realize the subject of zero-place verbs) are blocked from occuring as the complement of these verbs due to constraints on CASE (they are nominative, but these verbs select for an accusative NP complement).

Unlike transitive verbs, these verbs have a complement whose semantics is not linked to the semantics of the verbs. That is, in the lexical type for transitive verbs the INDEX of the complement is unified with the ARG2 feature of the verbal relation. By contrast, the lexical type for these verbs does not contain such a constraint, since the verbal relation does not have an ARG2 feature and the complement does not carry any semantics.

Another point to consider is that these inherent clitics behave syntactically just like reflexive pronouns. In particular, they have to agree with the subject. LXGram lacks a general implementation of binding phenomena; therefore this cannot be reused to force agreement between the subject and the complement. For the time being, the lexical type for these verbs constrains the AGR feature of the subject to be unified with the AGR feature of the complement. This feature is where information about person, number and gender is represented (see Section 5.10).

As a final note, inherent clitics cannot be part of long distance dependencies. Additionally, the complement is obligatory. Therefore, the verbs that select for an inherent pronoun complement constrain that complement to have a SYNSEM of the type *canonical-synsem*, as in LXGram this *synsem* type is incompatible with the *synsem* type of gaps (*gap*) and missing complements (*optional-synsem*).

9.1.6 Verbs with an inherent clitic complement and a PP complement

The type *verb-subj_np-inherent_clitic-prep-item* is the supertype of all lexical types for verbs with an inherent clitic and a PP complement. One of these types is the type *verb-inherent_clitic-comp_pp_com-lex*, for the verbs that take a PP complement headed by the preposition "com". One example is the verb "aconselhar-se (com alguém)" (*to get advice (from someone)*). There are types for verbs like this one, but selecting for PP complements with a different preposition.

The implementation of these types is similar to the implementation of the type for verbs with an inherent clitic and no other complement, described above in Section 9.1.5. They have an additional PP complement, and they are implemented as carrying a two-place relation in the semantics. The PP complement realizes the second argument of the semantic relation, and the preposition has no semantics.

9.1.7 Locative Complements

Some verbs select for a locative complement that can be realized by a variety of syntactically different constituents.

The following examples show that this complement can be introduced by different prepositions, different adverbs, and combinations of prepositions and adverbs.

(115) a. Moramos em Lisboa.
we live in Lisbon
We live in Lisbon.

b. Moramos sob a ponte.
we live under the bridge

We live under the bridge.

c. Moramos ali
we live there
We live there

d. Moramos perto.
we live nearby
We live nearby.

e. Moramos por ali.
we live PREPOSITION there
We live around there.

The lexical type *verb-intrans-obl_location-lex* is assigned to a verb like "morar" (*live*). This type constrains the complement of the verb to be a PP headed by the dummy preposition "em" (*in*). In the semantic representation there is a two-place predicate where the first argument corresponds to the subject of the verb, and the second argument corresponds to the PP complement. Figure 9.2 presents the MRS for the sentence in (115a).

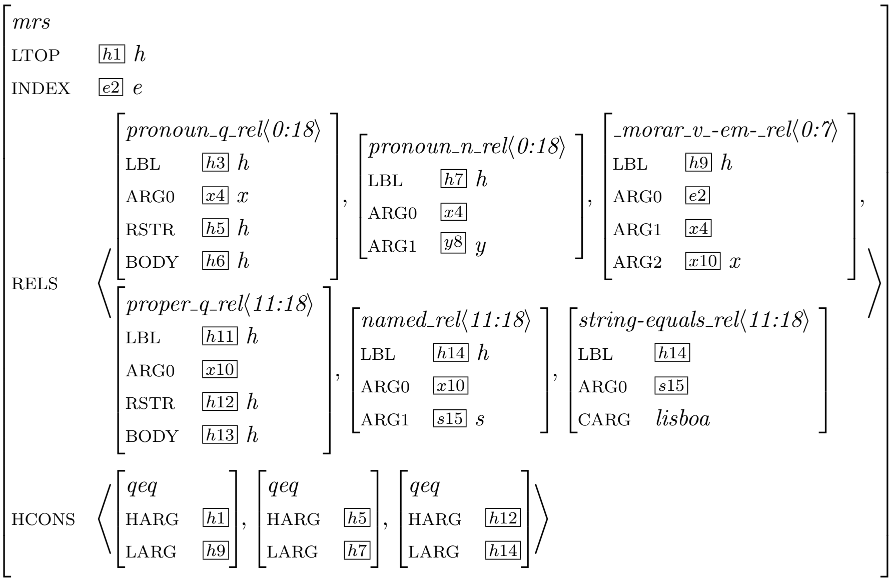

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:18⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:18⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_morar_v_-em-_rel⟨0:7⟩ LBL h9 h ARG0 e2 ARG1 x4 ARG2 x10 x
proper_q_rel⟨11:18⟩ LBL h11 h ARG0 x10 RSTR h12 h BODY h13 h
named_rel⟨11:18⟩ LBL h14 h ARG0 x10 ARG1 s15 s
string-equals_rel⟨11:18⟩ LBL h14 ARG0 s15 CARG lisboa
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h12 LARG h14

Figure 9.2: MRS for the sentence "Moramos em Lisboa" (*We live in Lisbon*).

In LXGram there is a specific syntax rule that is used to make locative adverbs appear as the complement of these verbs. Locative adverbs, like "ali" in the above examples, come in the lexicon with the semantics of modifiers. In order to make them appropriate to realize the second argument of the verbal relation of these locative verbs, a quantifier relation is added to the semantics in LXGram. Figure 9.3 shows the resulting MRS representation.

This additional quantifier relation is incorporated in the MRS representation by the syntax rule *np_semantics-adverb*, which takes as daughter a locative adverb. Locative adverbs are distinguished from the other adverbs in LXGram by a feature SEMSORT, which takes the value *location* for locative adverbs and a value subsumed by the type *non-location* for the other adverbs (these two types are incompatible). The *np_semantics-adverb* rule constrains the SEMSORT feature of the daughter to be *location*. The node produced by this rule has a HEAD type similar to that of dummy prepositions, and a PFORM with the value *em*: it is like a PP headed by a semantically vacuous preposition "em".

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{pronoun\_q\_rel}\langle\textit{0:11}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle\textit{0:11}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \begin{bmatrix} \textit{\_morar\_v\_-em-\_rel}\langle\textit{0:7}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{x10}\ x \end{bmatrix}, \begin{bmatrix} \textit{udef\_q\_rel}\langle\textit{8:11}\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h12}\ h \\ \text{BODY} & \boxed{h13}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_ali\_a\_rel}\langle\textit{8:11}\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{e15}\ e \\ \text{ARG1} & \boxed{x10} \end{bmatrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h12} \\ \text{LARG} & \boxed{h14} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 9.3: MRS for the sentence "Moramos ali" (*We live there*).

LXGram distinguishes these verbs from the verbs that take a non locative PP complement headed by "em" (like "insistir", *insist*) by positing different constraints on the SEMSORT feature of the complement in the different lexical types for these classes of verbs. A verb like "insistir" constrains the SEMSORT of its complement to be *non-location*, which prevents locative adverbs from appearing in that position.

Sentences like the one in (115b) are also accounted for. In this example, the locative complement is a PP, but it is introduced by a locative preposition that is not "em". It is desirable to make the MRS for a sentence like (115b) different from the MRS of a sentence like (115a), since the different prepositions carry different meanings. The solution that is implemented is similar to the solution for adverbs. A syntax rule applies to the PP (which is analyzed as having a preposition that carries semantics) that adds a quantifier to bind the variable that is the second argument of the verbal relation and the first argument of the relation for the preposition. The resulting MRS for the sentence in (115b) is in Figure 9.4.

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{matrix} \begin{bmatrix} \textit{pronoun\_q\_rel}\langle\textit{0:19}\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{pronoun\_n\_rel}\langle\textit{0:19}\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{y8}\ y \end{bmatrix}, \begin{bmatrix} \textit{\_morar\_v\_-em-\_rel}\langle\textit{0:7}\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{x10}\ x \end{bmatrix}, \\ \begin{bmatrix} \textit{udef\_q\_rel}\langle\textit{8:19}\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h12}\ h \\ \text{BODY} & \boxed{h13}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_sob\_p\_rel}\langle\textit{8:11}\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{e15}\ e \\ \text{ARG1} & \boxed{x10} \\ \text{ARG2} & \boxed{x16}\ x \end{bmatrix}, \begin{bmatrix} \textit{\_o\_q\_rel}\langle\textit{12:13}\rangle & \\ \text{LBL} & \boxed{h17}\ h \\ \text{ARG0} & \boxed{x16} \\ \text{RSTR} & \boxed{h18}\ h \\ \text{BODY} & \boxed{h19}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_ponte\_n\_1\_rel}\langle\textit{14:19}\rangle & \\ \text{LBL} & \boxed{h20}\ h \\ \text{ARG0} & \boxed{x16} \end{bmatrix} \end{matrix} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h12} \\ \text{LARG} & \boxed{h14} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h18} \\ \text{LARG} & \boxed{h20} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 9.4: MRS for the sentence "Moramos sob a ponte" (*We live under the bridge*).

The syntax rule used in such sentences (*np_semantics-pp*) is different from the syntax rule used when the complement of the locative verb is an adverb (*np_semantics-adverb*), because the *np_semantics-adverb* rule also applies to directional complements (see the next section). These two rules inherit from the same supertype as most of their constraints are shared.

Since only locative prepositions can appear in this context, the feature SEMSORT is also used to mark locative prepositions, and the *np_semantics-pp* syntax rule also constrains this feature.

For sentences like the one in (115e), it is necessary to allow locative adverbs to be the complement of some prepositions. Furthermore, it is necessary to allow the resulting constituent to be a complement of these verbs. In LXGram the prepositions that can have locative adverbs as complements have an additional lexical entry, with a type where the necessary constraints are implemented. This option is justified by the fact that only a few prepositions can take an adverbial complement. That is the case of prepositions like “até”, “de”, “desde”, “para”, “por”, for instance, while some locative prepositions, like “sob”, cannot have an adverbial complement. This lexical type is *preposition-predicational-comp_advp-lex*. The semantic features of this lexical type encode a relation for the preposition’s relation, as well as a quantifier binding the variable that instantiates the second argument of the preposition’s relation and the first argument of the adverb’s relation. The MRS for the sentence in (115e) is in Figure 9.5. The *np_semantics-pp* syntax rule also applies to the resulting PP.

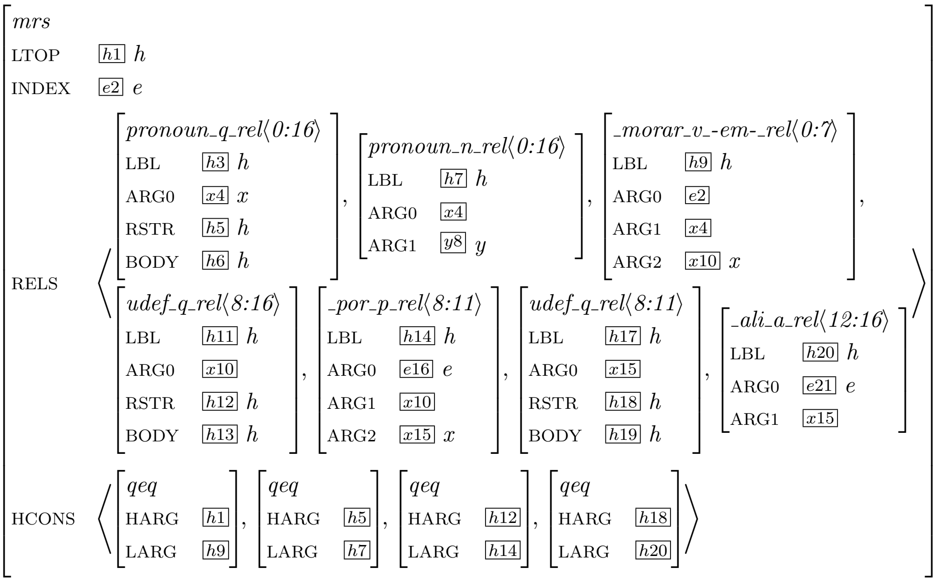

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:16⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:16⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_morar_v_-em-_rel⟨0:7⟩ LBL h9 h ARG0 e2 ARG1 x4 ARG2 x10 x
udef_q_rel⟨8:16⟩ LBL h11 h ARG0 x10 RSTR h12 h BODY h13 h
_por_p_rel⟨8:11⟩ LBL h14 h ARG0 e16 e ARG1 x10 ARG2 x15 x
udef_q_rel⟨8:11⟩ LBL h17 h ARG0 x15 RSTR h18 h BODY h19 h
_ali_a_rel⟨12:16⟩ LBL h20 h ARG0 e21 e ARG1 x15
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h12 LARG h14
qeq HARG h18 LARG h20

Figure 9.5: MRS for the sentence “Moramos por ali” (*We live around there*).

9.1.8 Directional Complements

Some verbs select for a complement that can be realized by a prepositional phrase (with a preposition taking an NP or adverb complement) or an adverbial phrase. The examples illustrate this:

(116) a. Íamos para Lisboa.
we were going to Lisbon
We were going to Lisbon.

b. Íamos a Lisboa.
we were going to Lisbon
We were going to Lisbon.

c. Íamos ali.
we were going there
We were going there.

d. Íamos para ali.
we were going to there
We were going there.

The analysis in LXGram for these sentences is similar to the analysis of sentences involving locative complements, presented in the previous section. These verbs select for a PP complement with a preposition that can be "a", "até" or "para". They also constrain the SEMSORT of their complement to be of type *direction*.

In the case of verbs with directional complements, the difference in the preposition used does not affect the meaning of the sentence. The two first sentences in the examples above are thus analyzed as containing a semantically vacuous preposition. The MRS representation is identical for both sentences, and it is presented in Figure 9.6.

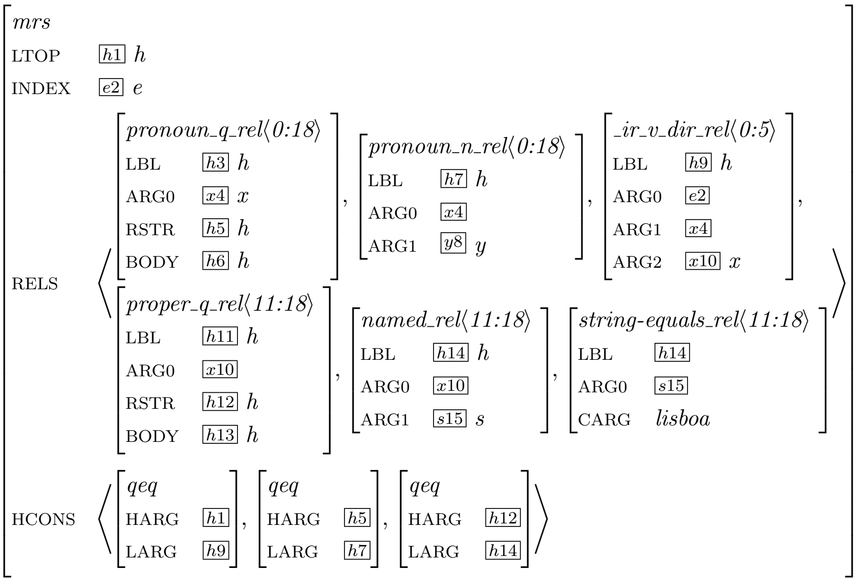

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:18⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:18⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_ir_v_dir_rel⟨0:5⟩ LBL h9 h ARG0 e2 ARG1 x4 ARG2 x10 x
proper_q_rel⟨11:18⟩ LBL h11 h ARG0 x10 RSTR h12 h BODY h13 h
named_rel⟨11:18⟩ LBL h14 h ARG0 x10 ARG1 s15 s
string-equals_rel⟨11:18⟩ LBL h14 ARG0 s15 CARG lisboa
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h12 LARG h14

Figure 9.6: MRS for the sentence "Íamos para Lisboa" (*We were going to Lisbon*).

The sentence in (116c) is analyzed with the *np_semantics-adverb* syntax rule applying to the adverb and produces the MRS shown in Figure 9.7.

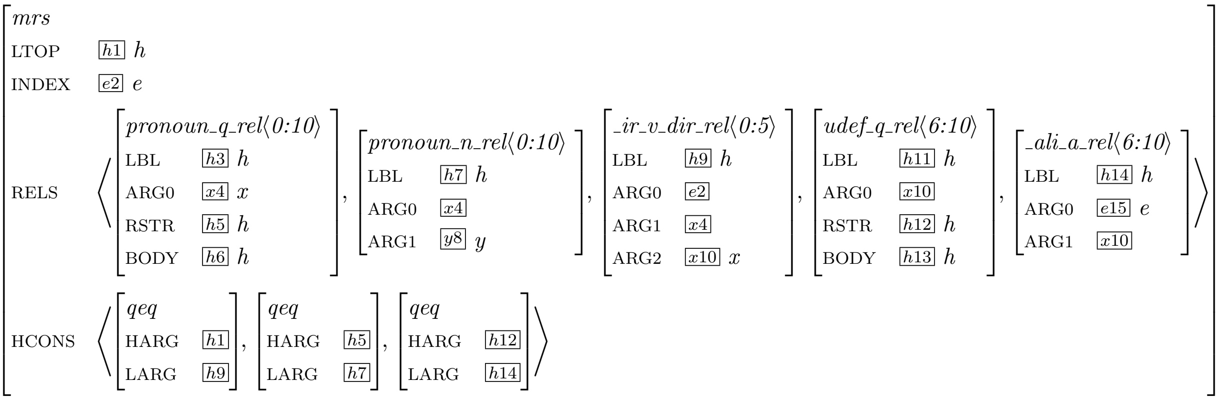

mrs
LTOP h1 h
INDEX e2 e
RELS
pronoun_q_rel⟨0:10⟩ LBL h3 h ARG0 x4 x RSTR h5 h BODY h6 h
pronoun_n_rel⟨0:10⟩ LBL h7 h ARG0 x4 ARG1 y8 y
_ir_v_dir_rel⟨0:5⟩ LBL h9 h ARG0 e2 ARG1 x4 ARG2 x10 x
udef_q_rel⟨6:10⟩ LBL h11 h ARG0 x10 RSTR h12 h BODY h13 h
_ali_a_rel⟨6:10⟩ LBL h14 h ARG0 e15 e ARG1 x10
HCONS
qeq HARG h1 LARG h9
qeq HARG h5 LARG h7
qeq HARG h12 LARG h14

Figure 9.7: MRS for the sentence "Íamos ali" (*We were going there*).

The exact way in which the feature SEMSORT is constrained in the *np_semantics-adverb* and the *np_semantics-pp* rules is the following. Both rules unify the SEMSORT of the daughter and the mother node. This feature is also constrained to be of the type *location-or-direction*, a supertype of the types *location* and *direction*. The *np_semantics-pp* further restricts it to be *location*.

Most adverbs cannot occur as the complement of either type of verb, and they have their SEMSORT feature constrained with the type *non-location-non-direction*, which is incompatible with *direction* and *location*. Adverbs like "aqui" (*here*) and "ali" (*there*), which can appear as locative complements and directional complements, have their SEMSORT feature underspecified.

Finally, a sentence like (116d), with a preposition and an adverb as the directional complement, produces an MRS exactly like the MRS in Figure 9.7, with just an adverb in the complement position. In this case there is no relation for the preposition. The prepositions that can head directional complements and take adverbs as complements have an additional lexical entry, with the lexical type *preposition-nonpredicational-comp_advp-lex*. Lexical items with this lexical type carry no relation for the preposition, but they include a quantifier to bind the variable that is the first argument of the adverb relation and the directional argument of the verb. The SEMSORT of the adverb complement is constrained to be *location*, but the SEMSORT of the preposition is *direction*. This allows for sequences of a preposition followed by a strictly locative adverb to appear as a directional complement, like "para antes do jardim" (*to a place before the garden*). Note that not all locative prepositions that can take an adverbial complement (the ones with a lexical entry of the type *preposition-predicational-comp_advp-lex*) have another lexical entry with the lexical type *preposition-nonpredicational-comp_advp-lex*. One example is "por", which can head locative complements, but not directional complements.

9.1.9 Clausal Subjects

In LXGram implements a few subcategorization frames for verbs that can take as subject an infinitive clause or a complementizer phrase. All these verbs can also take an NP as their subject, since pronouns can appear in the contexts where subject CPs can occur. Currently, there is one such lexical type for intransitive verbs and another one for transitive verbs. All other lexical types for verbs constrain the subject to be an NP. Both of these lexical types inherit from a supertype where the constraints are stated. This type constrains the HEAD feature of the subject appropriately (i.e. with a type that is the supertype of the HEAD type for nouns and the one for complementizers).

Infinitive subjects are handled with a unary syntax rule that takes as daughter a sentence with the head verb in the infinitive form and produces a node with a HEAD of the type *complementizer*. This syntax rule is called *infinitive-subject*. The rule is necessary in order to introduce in the semantics a *qeq* relation between the first argument of the relation for the main verb (the verb with the infinitive clause as subject) and the handle of the embedded infinitive.

9.2 Alternations

9.2.1 Passives

In LXGram, passives are handled by a lexical rule for participles that changes the elements of the SUBJ and COMPS lists. This rule is the *passive-participle* rule.

Passive participles can occur in passive constructions like the one in (117a), or they can be noun modifiers, as in (117b). They can also appear in absolute participial constructions (117c) and as the complement of some subordinating conjunctions and prepositions (117d), but these constructions are not implemented in LXGram yet.

(117) a. Anunciou que foi efectuada uma operação militar.
anounced that was conducted a operation military
He announced that a military operation was conducted.

b. Nos cinco anos decorridos desde a Guerra do Golfo, pouco mudou.
in the five years passed since the Gulf War little has changed
In the five years passed since the Gulf War, little has changed.

c. Accionada a bomba alternativa de combustível, o motor voltou a pegar.
activated the pump alternative of fuel the engine returned to start
After the alternative fuel pump was activated, the engine started again.

d. 5 horas depois de terminado o segundo ataque, ...
5 hours after finished the second attack
5 hours after the second attack was finished, ...

Since past participles (and gerunds) can be modifiers, they need to have a SELECT feature (see Section 5.3). Therefore, their HEAD type must inherit from the type *functor*. However, the other verbal forms cannot be modifiers, and they do not need to inherit from this type. In LXGram, verbs come in the lexicon with the HEAD value *verbal*. This type has subtypes like *passive-participle* and *gerund*, which also inherit from *functor*, and the subtype *verb*, which does not inherit from *functor*. The HEAD feature of verbs is constrained appropriately in the morphology rules: the *passive-participle* rule constrains it to be *past-participle*, the rule that produces gerunds constrains it to be *gerund*, and the remaining inflectional rules constrain it to be *verb*.

The subtypes *passive-participle* and *gerund* contain the constraints that control attachment possibilities for these elements (e.g. participles can only modify nouns, but gerunds can modify nouns and verbs).

The lexical rule for passive participles changes the valence features of the verb. It applies to verbs that take an NP subject and an accusative NP as their first complement and removes that element from the verb's COMPS list, by percolating only the tail of that list to the mother node. It also makes the SUBJ of the mother be the first complement of the daughter (it doesn't copy the entire *synsem*, because certain features, like CASE, have to be changed), and appends an element to the mother COMPS feature that is constrained to be a PP headed by "por" and whose INDEX is unified with the INDEX of the SUBJ of the daughter.

Figure 9.8 shows the MRS for a sentence in the passive voice.

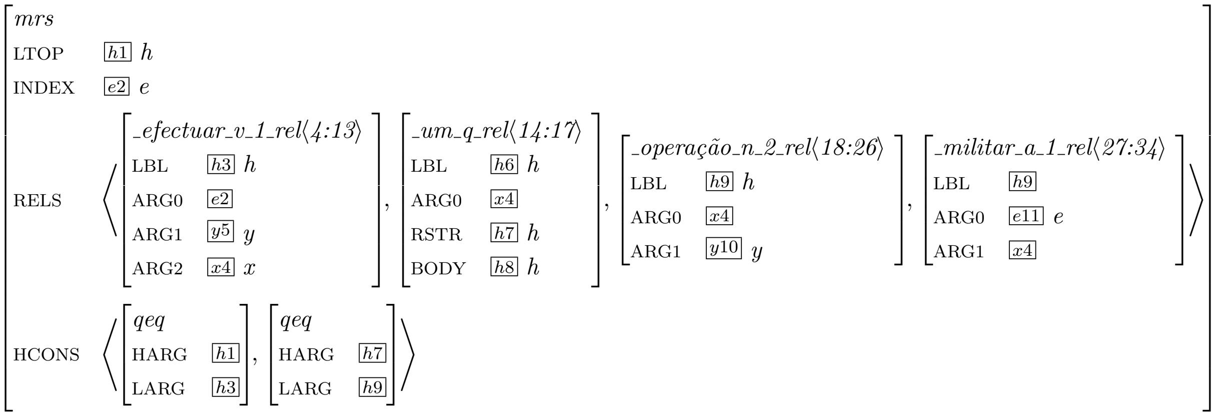


Figure 9.8: MRS for the sentence "Foi efectuada uma operação militar" (*A military operation was conducted*).

9.2.2 Anticausatives

LXGram implements the anticausative alternances shown in the following examples:

(118) a. A Maria acordou o Pedro.
the Maria woke up the Pedro
Maria woke up Pedro.

b. O Pedro acordou(*-se).
the Pedro woke up
Pedro woke up.

(119) a. A Maria assustou o Pedro.
the Maria scared the Pedro
Maria scared Pedro.

b. O Pedro assustou*(-se).
the Pedro scared CLITIC
Pedro got scared.

(120) a. A Maria acalmou o Perdo.
the Maria calmed down the Pedro
Maria calmed down Pedro.

b. O Pedro acalmou(-se).
the Pedro calmed down
Pedro calmed down.

The difference between these three pairs of sentences is the occurrence of an inherent clitic pronoun with the intransitive version of these verbs: with “acordar” (*wake up*) it is impossible, with “assustar” (*scare*) it is obligatory, and with “acalmar” (*calm down*) it is optional.

The intransitive version of these verbs is in the lexicon. Three different lexical types are used, reflecting the occurrence of inherent clitics. The transitive version is produced by the lexical rule *causative-alternation*. The verbs that can feed this rule inherit from the type *basic-verb-causative-alternation-elem*. The *causative-alternation* lexical rule constrains its daughter to be of this type (*basic-verb-causative-alternation-elem*).

This lexical rule produces a node whose complement is the subject of the daughter node, and has an additional subject. Currently, it adds a *cause_v_rel* relation to the semantics. The first argument of this relation is the added subject, the second argument is the relation of the daughter node. An example MRS is in Figure 9.9, for the sentence “A Maria assustou o Pedro”.

$$
\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textit{proper\_q\_rel}\langle 0{:}1\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix},
\begin{bmatrix} \textit{named\_rel}\langle 2{:}7\rangle & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \\ \text{ARG1} & \boxed{s8}\ s \end{bmatrix},
\begin{bmatrix} \textit{string-equals\_rel}\langle 2{:}7\rangle & \\ \text{LBL} & \boxed{h7} \\ \text{ARG0} & \boxed{s8} \\ \text{CARG} & \textit{maria} \end{bmatrix},
\begin{bmatrix} \textit{cause\_v\_rel}\langle 8{:}16\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{h10}\ h \end{bmatrix}, \\
\begin{bmatrix} \textit{\_assustar-se\_v\_rel}\langle 8{:}16\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{e12}\ e \\ \text{ARG1} & \boxed{x13}\ x \end{bmatrix},
\begin{bmatrix} \textit{proper\_q\_rel}\langle 17{:}18\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{RSTR} & \boxed{h15}\ h \\ \text{BODY} & \boxed{h16}\ h \end{bmatrix},
\begin{bmatrix} \textit{named\_rel}\langle 19{:}24\rangle & \\ \text{LBL} & \boxed{h17}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{ARG1} & \boxed{s18}\ s \end{bmatrix},
\begin{bmatrix} \textit{string-equals\_rel}\langle 19{:}24\rangle & \\ \text{LBL} & \boxed{h17} \\ \text{ARG0} & \boxed{s18} \\ \text{CARG} & \textit{pedro} \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h9} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h10} \\ \text{LARG} & \boxed{h11} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h15} \\ \text{LARG} & \boxed{h17} \end{bmatrix}
\right\rangle
\end{bmatrix}
$$

Figure 9.9: MRS for the sentence “A Maria assustou o Pedro” (*Maria scared Pedro*).

Chapter 10

Coordination

LXGram's implementation of coordination is based on the coordination module of the LinGO Grammar Matrix. There are sets of rules for different types of constituents: noun phrases, nouns, verb phrases, verbs, sentences, nonpredicational prepositional phrases and for coordination of modifiers and predicates.

In each set of rules there are three rules: a top coordination rule, a middle coordination rule and a bottom coordination rule. There are also left coordination rules (for discontinuous conjunctions, discussed in Section 10.2) and special bottom rules for asyndetic coordination (presented below in Section 10.3).

All of these rules are binary. Bottom coordination rules have as daughters a conjunction and a phrase of the coordinated category. Top and mid coordination rules have as their left daughter a phrase of the coordinated category, and as their right daughter they have a middle or a bottom coordination rule. This is controlled with the feature SYNSEM|LOCAL|CAT|COORDINATION. So, for instance, top coordination rules constrain their right daughter to have this feature with the value *bottom-or-mid-coordination*.

Each middle coordination rule adds a conjunction relation to the MRS, with the same name as the relation introduced by the conjunction in the bottom coordination rule (this information is percolated up the syntax tree). Top rules do not add semantics.

Left coordination rules have as their left daughter a conjunction, and a coordinated phrase as their right daughter, just like bottom rules. They can only occur as the left daughter of a top or middle coordination rule. They are used to handle discontinuous conjunction, like "ou ... ou ..." (*either ... or ...*). The conjunctions that appear as the left daughter of left coordination rules carry no semantics, and they need dedicated entries in the lexicon.

In the cases of NP coordination (and, in LXGram, also coordination of non-predicational PPs), the top and middle rules also add quantifier relations, just like in the LinGO Grammar Matrix (but see the treatment of modification of coordinated NPs in Section 10.5). More on this is in Section 10.4.

The following sentence illustrates the use of top, middle and bottom rules for coordination:

(121) É um álbum cru, forte e rápido.
is an album raw strong and fast
It is a raw, strong and fast album.

The syntax tree for this sentence is in Figure 10.2.

In this tree, the node spanning "cru, forte e rápido" is produced with the top coordination rule for modifier coordination, the node spanning "forte e rápido" is obtained with the middle rule, and the node spanning "e rápido" comes from the bottom rule. The semantic representation

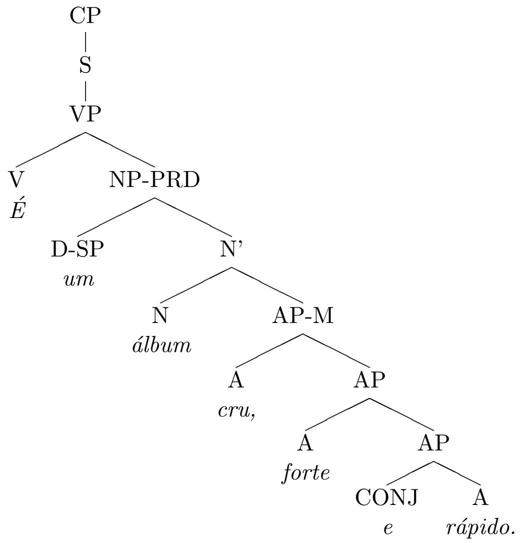


Figure 10.1: Syntax tree for the sentence “É um álbum cru, forte e rápido” (*It’s a raw, strong and fast album*).

contains two binary coordination relations. For instance, the bit of semantics corresponding to the node produced by the top coordination rule is the following (where x_{10} is the referential index of the modified noun):

$$\_e\_coord(e_{19},\ \_cru\_a(e_{17}, x_{10}),\ \_e\_coord(e_{18},\ \_forte\_a(e_{22}, x_{10}),\ \_rápido\_a(e_{23},\ x_{10})))$$

The relation for the conjunction is the second *_e_coord* relation. The first *_e_coord* relation is introduced in the middle coordination rule, and the predicate name is copied from the predicate name of the relation introduced by the conjunction.

10.1 Coordination of Sentences and Verb Phrases

If unrestricted coordination of sentences and verb phrases is allowed, there will be multiple analyses for sentences like the following one:

(122) Cheguei, vi e venci.
I arrived I saw and I conquered
I came, I saw, I conquered.

The problem here is that this sentence can be analyzed as a coordination of sentences, all of them with null subjects, or as a coordination of VPs with a null subject. In LXGram we restrict coordination of VPs to apply only to infinitives and gerunds. The sentence in this example is parsed as involving a coordination of sentences.

10.2 Discontinuous Conjunctions

In order to handle discontinuous conjunctions, illustrated in the following example, there are also left coordination rules in LXGram.

(123) É um álbum não só forte mas também rápido.
is an album not only strong but also fast
It is a both strong and fast album.

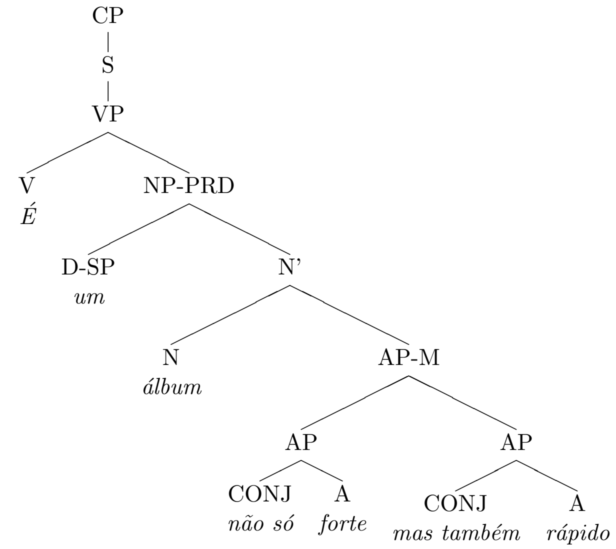


Figure 10.2: Syntax tree for the sentence "É um álbum não só forte mas também rápido" (*It's a both strong and fast album*).

Left coordination rules are binary rules whose left daughter is a conjunction carrying no semantics and whose right daughter is a coordinated phrase. They can only occur as the left daughter of top and middle coordination rules.

The bit of semantics spanning "não só forte mas também rápido" is:

$$\textit{\_não+só+mas+também\_coord}(e_{19},\ \textit{\_forte\_a}(e_{17},\ x_{10}),\ \textit{\_rápido\_a}(e_{18},\ x_{10}))$$

The *_não+só+mas+também_coord* is introduced by the lexical item for "mas também", and the item for "não só" has empty semantics.

Since the predicate name of the conjunction used in the bottom rule is percolated up the coordination phrases and visible in all coordination rules, the co-occurrence of elements like "não só" and "mas também" is controlled in this way. So for instance, we can allow these sequences, as well as sequences like "ou ... ou ...", but block ungrammatical sequences like "não só ... ou ...". The bottommost coordination relation is encoded in the feature SYNSEM|LOCAL|CONT|KEYS|COORD-REL, which is percolated by the coordination rules and where the predicate name can be obtained under the feature PRED. Phrases not involving coordination, as well as top coordination phrases, have the type *no-relation* as the value of this feature, which is incompatible with the type for actual semantic relations.

10.3 Asyndetic Coordination

The sentence in (124) illustrates the possibility of coordination with no conjunction.

(124) É um álbum forte, rápido.
is an album strong fast
It is a strong, fast album.

LXGram's implementation assumes the missing conjunction has the same samantics as "e" (*and*).

Asyndetic coordination is implemented by having unary versions of the bottom rules. The asyndetic bottom coordination rules take a phrase as daughter and add a semantic relation corresponding to the missing conjunction.

10.4 NP Coordination

LXGram implements an analysis of NP coordination similar to the analysis of the LinGO Grammar Matrix and the LinGO English Resource Grammar. In these analyses, quantifier relations are added into the semantics in order to provide a referential index for the entire coordination.

Figure 10.3 shows the MRS for the sentence in (125). NP coordination gives rise to MRS representations with a relation for the conjunction, in this case the *_e_coord_rel* relatio, which comes from the lexical entry for the conjunction. This relation takes as arguments the referential indices of the coordinated NPs. It also introduces another referential index, x_8 in this example, in its C-ARG feature. This referential index is the argument of the verbal relation corresponding to its subject, since the subject of that sentence is the NP coordination. It is bound by the *udef_q_rel* present in that MRS. In the LinGO Grammar Matrix, this quantifier that binds the variable of the conjunction relation is introduced in the top and middle rules for NP coordination. In LXGram this has been slightly changed, in order to accomodate modification of coordinated NPs (see Section 10.5).

(125) Não votaram Portugal e Espanha.
not voted Portugal and Spain
Portugal and Spain didn't vote.

$$
\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textit{neg\_rel}\langle 0{:}3\rangle & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e5}\ e \\ \text{ARG1} & \boxed{h4}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_votar\_v\_-em-\_rel}\langle 4{:}11\rangle & \\ \text{LBL} & \boxed{h6}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{x8}\ x \\ \text{ARG2} & \boxed{y7}\ y \end{bmatrix},
\begin{bmatrix} \textit{udef\_q\_rel}\langle 12{:}30\rangle & \\ \text{LBL} & \boxed{h9}\ h \\ \text{ARG0} & \boxed{x8} \\ \text{RSTR} & \boxed{h10}\ h \\ \text{BODY} & \boxed{h11}\ h \end{bmatrix}, \\
\begin{bmatrix} \textit{proper\_q\_rel}\langle 12{:}20\rangle & \\ \text{LBL} & \boxed{h12}\ h \\ \text{ARG0} & \boxed{x13}\ x \\ \text{RSTR} & \boxed{h14}\ h \\ \text{BODY} & \boxed{h15}\ h \end{bmatrix},
\begin{bmatrix} \textit{named\_rel}\langle 12{:}20\rangle & \\ \text{LBL} & \boxed{h16}\ h \\ \text{ARG0} & \boxed{x13} \\ \text{ARG1} & \boxed{s17}\ s \end{bmatrix},
\begin{bmatrix} \textit{string-equals\_rel}\langle 12{:}20\rangle & \\ \text{LBL} & \boxed{h16} \\ \text{ARG0} & \boxed{s17} \\ \text{CARG} & \textit{portugal} \end{bmatrix},
\begin{bmatrix} \textit{\_e\_coord\_rel}\langle 21{:}22\rangle & \\ \text{LBL} & \boxed{h18}\ h \\ \text{C-ARG} & \boxed{x8} \\ \text{L-INDEX} & \boxed{x13} \\ \text{R-INDEX} & \boxed{x19}\ x \end{bmatrix}, \\
\begin{bmatrix} \textit{proper\_q\_rel}\langle 23{:}30\rangle & \\ \text{LBL} & \boxed{h20}\ h \\ \text{ARG0} & \boxed{x19} \\ \text{RSTR} & \boxed{h21}\ h \\ \text{BODY} & \boxed{h22}\ h \end{bmatrix},
\begin{bmatrix} \textit{named\_rel}\langle 23{:}30\rangle & \\ \text{LBL} & \boxed{h23}\ h \\ \text{ARG0} & \boxed{x19} \\ \text{ARG1} & \boxed{s24}\ s \end{bmatrix},
\begin{bmatrix} \textit{string-equals\_rel}\langle 23{:}30\rangle & \\ \text{LBL} & \boxed{h23} \\ \text{ARG0} & \boxed{s24} \\ \text{CARG} & \textit{espanha} \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h3} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h4} \\ \text{LARG} & \boxed{h6} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h10} \\ \text{LARG} & \boxed{h18} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h14} \\ \text{LARG} & \boxed{h16} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h21} \\ \text{LARG} & \boxed{h23} \end{bmatrix}
\right\rangle
\end{bmatrix}
$$

Figure 10.3: MRS for the sentence "Não votaram Portugal e Espanha" (*Portugal and Spain didn't vote*).

10.5 Modification of Coordinated Noun Phrases

A sentence like the one in (126) shows a relative clause modifying the coordination of noun phrases ("a paternidade e a sexualidade"), and not just the last noun phrase (note that the verb form inside the relative clause exhibits plural morphology).

(126) Em questão estiveram a paternidade e a sexualidade que condicionam
in question were the paternity and the sexuality that condition.PLURAL
a vida familiar.
the life familiar
Under debate were the paternity and sexuality that condition family life.

Figure 10.4 presents the syntax tree obtained with LXGram for this sentence. In order to accomodate sentences like this one, the top rule for NP coordination does not produce a full NP, but rather an N' constituent, visible in that tree. This node has the MARKING type *n-marking* (see Section 8.6). The quantifier relation that is added in top coordination rules in the LinGO Grammar Matrix is instead added by the rule for bare NPs in LXGram (note that it is the same relation, a *udef_q_rel* relation).

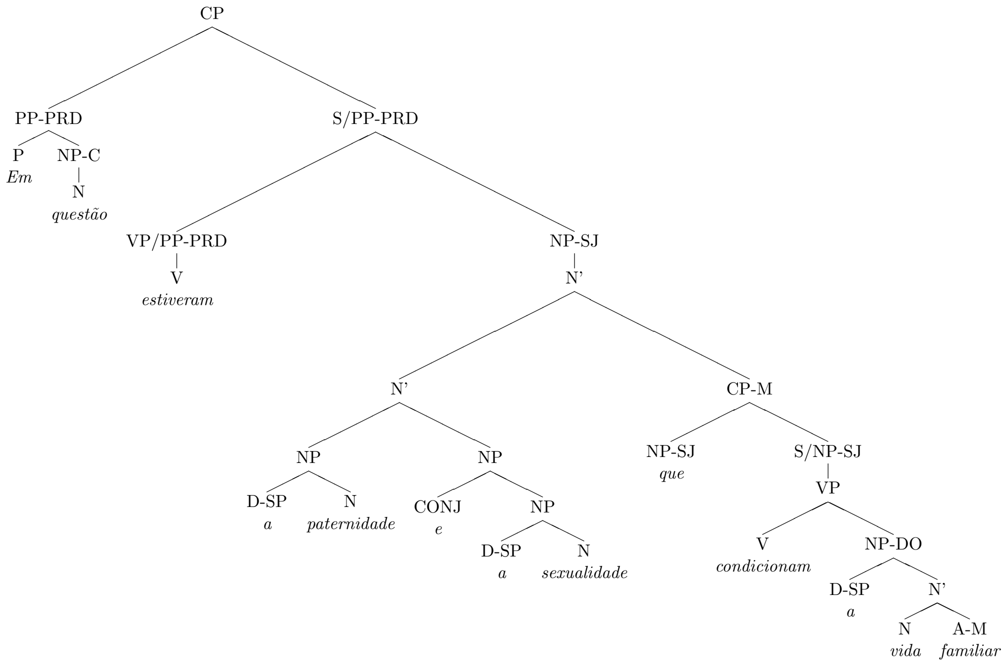


Figure 10.4: Syntax tree for the sentence "Em questão estiveram a paternidade e a sexualidade que condicionam a vida familiar" (*Under debate were the paternity and sexuality that condition family life*).

The N' node that is produced by the top coordination rule for NP coordination is also constrained in such a way that no overt material can precede it within the same NP (i.e. before the rule for bare NPs applies). This is restricted with SYNSEM|LOCAL|CONT|KEYS|QUANT-REL constrained to be the quantifier of a bare NP (see Section 8.5.2 and Section 8.6.4.1 for explanations of how this feature is used in LXGram), and with the feature SYNSEM|LOCAL|CAT|MODIFICATION |MODIFIABLE constrained to be of the type *right-modifiable*, which allows modification of this constituent only on the right side (Section 5.12 explains the use of this feature).

10.6 Sentence Initial Conjunctions

The possibility of coordinative conjunctions in sentence initial position has not been implemented yet.

Chapter 11

Long Distance Dependencies

Long distance dependencies involve a fronted constituent that can be arbitrarily distant from the head that selects it (if it is the subject or a complement of that head) or the head that they modify. Some examples are in (127), with the fronted constituent between brackets and a dash (-) in the place where that constituent would be expected (i.e. the gap). The examples are composed of a relative clause, a sentence with a topicalized element, and a wh-question.

(127) a. O livro [de que] nós falámos - esgotou.
the book of which we spoke sold out
The book we told you about sold out.

b. [Desse incidente] prometo que falaremos - depois.
about that incident I promise that we will talk later
About that incident I promise we will talk later.

c. [Quando] é que disseste que Nelson Mandela nasceu - ?
when you said that Nelson Mandela was born
When did you say Nelson Mandela was born?

The main characteristic of long distance dependencies is that there is no theoretical upper bound to the distance between the dislocated element and the gap:

(128) a. [Quem] é que - matou César?
who killed Caesar
Who killed Caesar?

b. [Quem] é que disseste que - matou César?
who you said that killed Caesar
Who did you say killed Caesar?

c. [Quem] é que disseste que se pensa que - matou César?
who you said that SUBJ thinks that - killed Caesar
Who did you say is thought to have killed Caesar?

d. [Quem] é que disseste que esse historiador referiu que se pensa que -
who you said that that historian mentioned that SUBJ thinks that
matou César ?
killed Caesar
Who did you say that historian mentioned is thought to have killed Caesar?

The implementation of unbounded dependencies in LXGram follows the typical HPSG analyses of this phenomenon and is essencially the implementation in the LinGO Grammar Matrix.

Three features under the NON-LOCAL attribute of synsems control unbounded dependencies: SLASH, QUE, and REL. The feature SLASH is a set of objects of the type *local* (the value of the feature LOCAL of synsems), and encodes information about gaps. The feature REL is a set of indices, and it is used to account for pied-piping in relative clauses (as in the example in (127a), where the preposition "de" is dislocated together with the relative pronoun "que"). The feature QUE is also a set of indices, and it is used to account for pied-piping in wh-question (as in *De que estás a falar?* "What are you talking about?") . These sets are implemented as difference lists.

In LXGram, the list valued feature ARG-ST of lexical items is the concatenation of the valence lists SUBJ and COMPS of that lexical item. ARG-ST is a list of synsems, like SUBJ and COMPS. This feature ARG-ST is used for SLASH amalgamation: the SLASH of a lexical item is the union of the SLASH sets of all the elements of its ARG-ST. The other NON-LOCAL features, QUE and REL, are also amalgamated this way. An example of SLASH amalgamation follows, for a transitive verb:

$$\begin{bmatrix} \text{SYNSEM} & \begin{bmatrix} \text{LOCAL}|\text{CAT}|\text{VAL} & \begin{bmatrix} \text{SUBJ} & \left\langle \boxed{1} \right\rangle \\ \text{COMPS} & \left\langle \boxed{2} \right\rangle \end{bmatrix} \\ \text{NON-LOCAL} & \begin{bmatrix} \text{SLASH} & \boxed{A} \cup \boxed{B} \\ \text{QUE} & \boxed{C} \cup \boxed{D} \\ \text{REL} & \boxed{E} \cup \boxed{F} \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \left\langle \boxed{1} \begin{bmatrix} \text{NON-LOCAL} & \begin{bmatrix} \text{SLASH} & \boxed{A} \\ \text{QUE} & \boxed{C} \\ \text{REL} & \boxed{E} \end{bmatrix} \end{bmatrix}, \boxed{2} \begin{bmatrix} \text{NON-LOCAL} & \begin{bmatrix} \text{SLASH} & \boxed{B} \\ \text{QUE} & \boxed{D} \\ \text{REL} & \boxed{F} \end{bmatrix} \end{bmatrix} \right\rangle \end{bmatrix}$$

The analysis of unbounded dependencies can be outlined as follows. Lexical items have empty sets in their SLASH feature, but an element is added to this set at the point where a gap is expected. Higher in the tree, an element in the SLASH set is removed from it at the same time as a constituent is projected in the left periphery of a clause. The grammar's start symbol is constrained to have an empty SLASH: all gaps that are introduced must be removed from SLASH at some point.

This analysis is slightly different from the theoretical HPSG analysis, but it is equivalent. The differences have to do with limitations of the system where LXGram is implemented. The first difference is that sets have to be implemented as feature structures (namealy as difference lists), which means that we cannot test for membership, but only check whether a particular element is the first element of a difference list. We also cannot remove an arbitrary element from a difference list, only an element occurring in a specific position in that list. Also, because we do not have access to general operations on sets and lists, syntax or lexical rules have to be used to introduce gaps, instead of just employing a general constraint on lexical types to ensure that every syntactic dependent appears in a valence list or in the SLASH. Additionally, for efficiency reasons, in LXGram SLASH is constrained to have at most one element, even though there are grammatical (albeit rare) examples of multiple extraction. In this text, we may nevertheless talk about set union (but we will mean difference list appends), set membership and set difference.

11.1 Gap Introducing Rules

Gaps are introduced through a set of unary rules. There are rules for gaps in subject position, complement position, and adjunct position. The rules for gaps in subject or complement position constrain one element of SUBJ or COMPS of the daughter node to be a synsem of the type *gap*. The corresponding valence lists of the mother node have this element removed.

The type *gap* is a subtype of *synsem* and in its definition is the constraint that its LOCAL object is an element of its SLASH set (and its only element):

$$\begin{bmatrix} \textit{gap} & \\ \textsc{local}\ \boxed{1} & \\ \textsc{non-local} & \begin{bmatrix} \textsc{slash} & \left\{\boxed{1}\right\} \\ \textsc{que} & \{\} \\ \textsc{rel} & \{\} \end{bmatrix} \end{bmatrix}$$

The type *gap* is incompatible with some other subtypes of *synsem*, most notably the type *canonical-synsem*. Elements that cannot be extracted are thus constrained to have a SYNSEM of this type, *canonical-synsem*. For instance, this is the case of the complement of prepositions (that is, prepositions constrain the element in their COMPS feature to be a *canonical-synsem*). This constraint blocks the possibility of stranded prepositions in Portuguese (unlike English, as can be seen in the example above (127a)).

Most rules simply percolate the NON-LOCAL of their head daughter. The rules that introduce gaps do it, too.

Gaps are discharged higher in the tree by filler constructions (see Section 11.2, Section 11.3 and Section 11.6).

The unary syntax rules that introduce gaps are the rules named *subject-head_extracted*, *head-complement_extracted*, *head-functor-isect_extracted*, *head-functor-isect_wh_extracted*, *head-functor-scopal_extracted* and *head-functor-scopal_wh_extracted*.

The first two rules are responsible for introducing a gap in subject and complement position, respectively. The latter four introduce gaps in adjunct position. The rules *head-functor-isect_extracted* and *head-functor-isect_wh_extracted* are for intersective modifiers, and *head-functor-scopal_extracted* and *head-functor-scopal_wh_extracted* for scopal modifiers. The difference between the rule *head-functor-isect_extracted* and the rule *head-functor-isect_wh_extracted* is that the second one is used in wh-questions, whereas the first one is used for topicalizations and relative clauses. The difference between the rule *head-functor-scopal_extracted* and the rule *head-functor-scopal_wh_extracted* is analogous.

Both *subject-head_extracted* and *head-complement_extracted* inherit from the Matrix type *head-valence-phrase*, where the NON-LOCAL features of the head daughter are constrained to percolate to the mother node. For the other syntax rules, introducing a gap in adjunct position, the SLASH feature of the mother node is the union of the SLASH feature of the daughter with the set containing the LOCAL of the extracted adjunct.

As an example, here are the main constraints on *head-complement_extracted*, the syntax rule responsible for introducing a gap as the first complement of an element:

$$\begin{bmatrix} \textit{head-complement\_extracted} & \\ \text{SYNSEM} & \begin{bmatrix} \text{LOCAL}|\text{CAT} & \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{VAL} & \begin{bmatrix} \text{SUBJ} & \boxed{2} \\ \text{COMPS} & \boxed{3} \end{bmatrix} \end{bmatrix} \\ \text{NON-LOCAL}\ \boxed{4} & \end{bmatrix} \\ \text{HEAD-DTR}|\text{SYNSEM} & \begin{bmatrix} \text{LOCAL}|\text{CAT} & \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{VAL} & \begin{bmatrix} \text{SUBJ} & \boxed{2} \\ \text{COMPS} & \begin{bmatrix} \text{FIRST} & \textit{gap} \\ \text{REST} & \boxed{3} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \text{NON-LOCAL}\ \boxed{4} & \end{bmatrix} \end{bmatrix}$$

For instance, when this rule is applied to a transitive verb, the following structureis obtained, due to SLASH amalgamation and the constraints on *gap*:

$$\begin{array}{c} \text{VP} \\ \begin{bmatrix} \text{SYNSEM} & \begin{bmatrix} \text{LOCAL}|\text{CAT}|\text{VAL} & \begin{bmatrix} \text{SUBJ}\ \left\langle \boxed{1} \right\rangle \\ \text{COMPS}\ \langle\rangle \end{bmatrix} \\ \text{NON-LOCAL}|\text{SLASH}\ \boxed{\mathit{A}} \cup \left\{ \boxed{3} \right\} & \end{bmatrix} \end{bmatrix} \\ | \\ \text{V} \\ \begin{bmatrix} \text{SYNSEM} & \begin{bmatrix} \text{LOCAL}|\text{CAT}|\text{VAL} & \begin{bmatrix} \text{SUBJ}\ \left\langle \boxed{1} \right\rangle \\ \text{COMPS}\ \left\langle \boxed{2} \begin{bmatrix} \textit{gap} \\ \text{LOCAL}\ \boxed{3} \\ \text{NON-LOCAL}|\text{SLASH}\ \left\{ \boxed{3} \right\} \end{bmatrix} \right\rangle \end{bmatrix} \\ \text{NON-LOCAL}|\text{SLASH}\ \boxed{\mathit{A}} \cup \left\{ \boxed{3} \right\} & \end{bmatrix} \\ \text{ARG-ST} & \left\langle \boxed{1} \begin{bmatrix} \text{NON-LOCAL}\ \boxed{\mathit{A}} \end{bmatrix}, \boxed{2} \right\rangle \end{bmatrix} \end{array}$$

11.2 Relative Clauses

Relative clauses are modeled via a syntax rule that takes two daughters. The right daughter is a sentence with a non-empty SLASH, and the left daughter's LOCAL is unified with the element of the right daughter's SLASH. The mother node has an empty SLASH and it is constrained to be a noun modifier (this construction is non-headed).

The filler daughter (i.e. the left one) is additionally constrained to have a non-empty NON-LOCAL|REL, and the index in this REL singleton set is unified with the index of the modified noun.

The only lexical items with a non-empty REL and an empty ARG-ST are relative constituents, so a filler must contain one of these. Also, the feature NON-LOCAL|REL is also amalgamated in terminal nodes like SLASH is, and percolated in syntax rules in a similar way.

The rule for relative clauses is called *clause_relative*. Figure 11.1 shows the relevant features at work.

Figure 11.2 shows the MRS for an NP containing a relative clause. The relative pronoun carries no semantics. The INDEX of the modified noun is unified with the INDEX of the relative

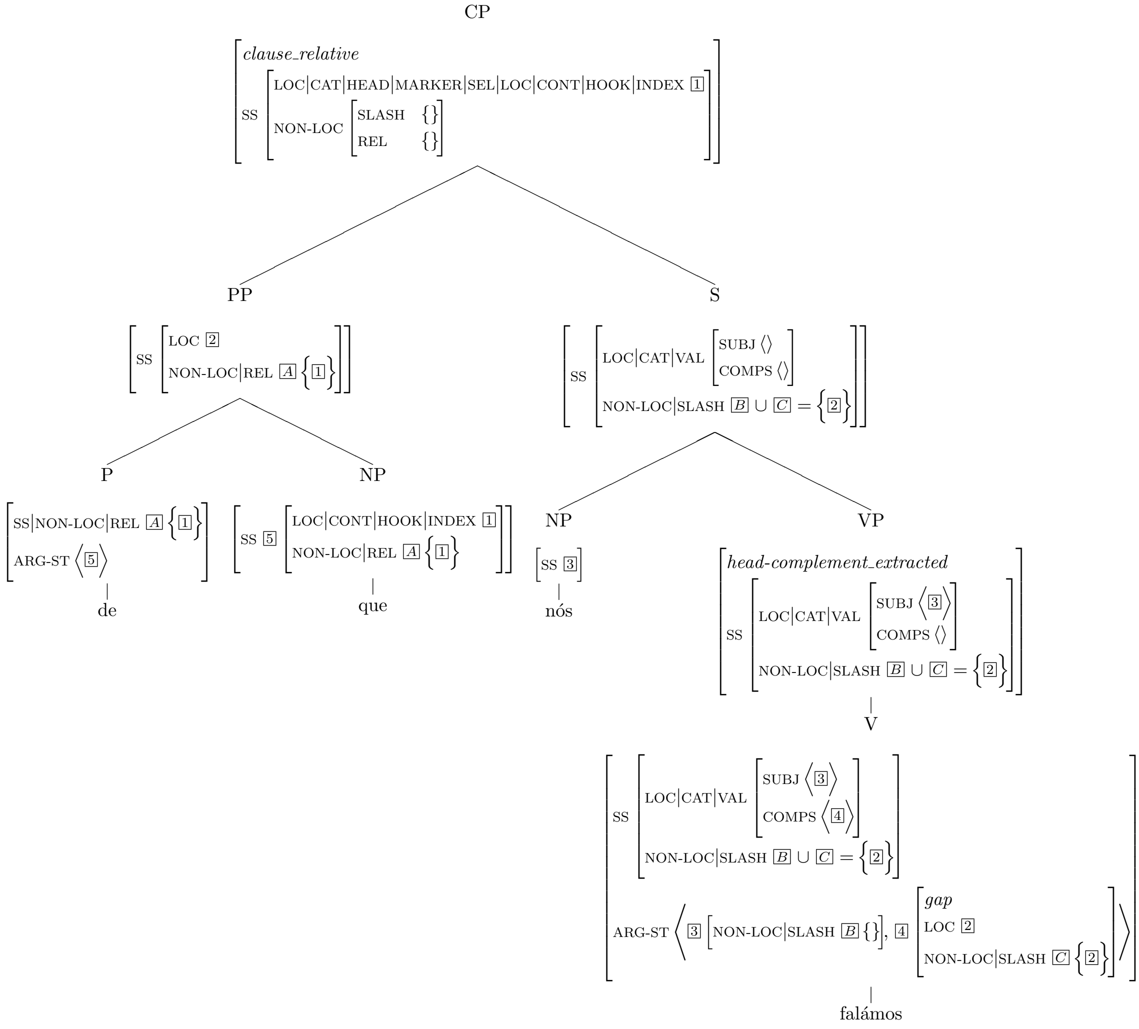


Figure 11.1: Example of a relative clause. The clause is “de que nós falámos” (*about which/whom we spoke*).

$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{mrs}} \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{x2}\ x \\
\text{RELS} & \left\langle \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_o\_q\_rel}\langle 0{:}1\rangle} \\ \text{LBL} & \boxed{h1} \\ \text{ARG0} & \boxed{x2} \\ \text{RSTR} & \boxed{h3}\ h \\ \text{BODY} & \boxed{h4}\ h\end{array}\right], \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_livro\_n\_rel}\langle 2{:}7\rangle} \\ \text{LBL} & \boxed{h5}\ h \\ \text{ARG0} & \boxed{x2}\end{array}\right], \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{pronoun\_q\_rel}\langle 15{:}18\rangle} \\ \text{LBL} & \boxed{h6}\ h \\ \text{ARG0} & \boxed{x7}\ x \\ \text{RSTR} & \boxed{h8}\ h \\ \text{BODY} & \boxed{h9}\ h\end{array}\right], \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{pronoun\_n\_rel}\langle 15{:}18\rangle} \\ \text{LBL} & \boxed{h10}\ h \\ \text{ARG0} & \boxed{x7} \\ \text{ARG1} & \boxed{y11}\ y\end{array}\right], \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{\_falar\_v\_-de-\_rel}\langle 19{:}26\rangle} \\ \text{LBL} & \boxed{h5} \\ \text{ARG0} & \boxed{e12}\ e \\ \text{ARG1} & \boxed{x7} \\ \text{ARG2} & \boxed{x2}\end{array}\right] \right\rangle \\
\text{HCONS} & \left\langle \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{qeq}} \\ \text{HARG} & \boxed{h3} \\ \text{LARG} & \boxed{h5}\end{array}\right], \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{qeq}} \\ \text{HARG} & \boxed{h8} \\ \text{LARG} & \boxed{h10}\end{array}\right] \right\rangle
\end{array}\right]$$

Figure 11.2: Example MRS with a relative clause. The MRS corresponds to the NP "o livro de que nós falámos" (*the book that we spoke about*).

pronoun, as already seen in Figure 11.1. In this example, the relative pronoun is in the prepositional complement of the verb, and as a result, the INDEX of the noun is the second argument of the verbal relation.

11.2.1 Free Relative Clauses

In LXGram there is a rule for free relative clauses, called *free-relative-clause.* It accounts for relative clauses without an expressed antecedent, like the example in (129)

(129)	[Quem	não	tem	cão]	caça	com	gato.
	who	not	has	dog	hunts	with	cat

Who doesn't own a dog goes hunting with a cat.

This rule is similar to the rule *clause_relative* presented above. The main difference is that instead of producing a noun modifier, *free-relative-clause* produces a noun phrase (the mother node has a HEAD feature of the type *noun* and it is saturated). Figure 11.3 shows the analysis for the free relative clause in (129).

In free relative clauses, semantics has to be provided for the missing antecedent. The extra bit of semantics can be put in the syntax rule or in the relative pronouns (as they are typically different from the relative pronouns that appear in relative clauses with an expressed antecedent). In view of the fact that the relative pronouns that can appear in this construction seem to carry some semantics (e.g. *who* vs. *what*), the extra semantic relations are put in the lexical entry of the relative pronouns that appear in free relative clauses.

Figure 11.4 shows the MRS representation for the relative clause in (129). The first two relations are from the lexical entry for "quem".

There needs to be a way to control rule application (*clause_relative* vs. *free-relative-clause*) according to the relative element that is present in the filler daughter, because the relative constituents that can appear in free relative clauses are different from the ones that can appear in relative clauses with an expressed antecedent.

In LXGram, this is handled by making the feature REL not a set of indices but a set of structures with two features: a feature where an index is stored and a boolean feature, FREE, that encodes whether a free relative clause is possible. An example follows. The relative pronoun "quem" (*who*) can be used in both kinds of relative clauses, but if the relative antecedent is expressed, "quem" can only be used if it is the complement of a preposition. Accordingly, there are two lexical entries for "quem": one where the element in its REL feature has FREE +, and another one where this feature is - but where the CASE feature under HEAD is also constrained to be *oblique* (the case exhibited by NPs that are complements of prepositions).

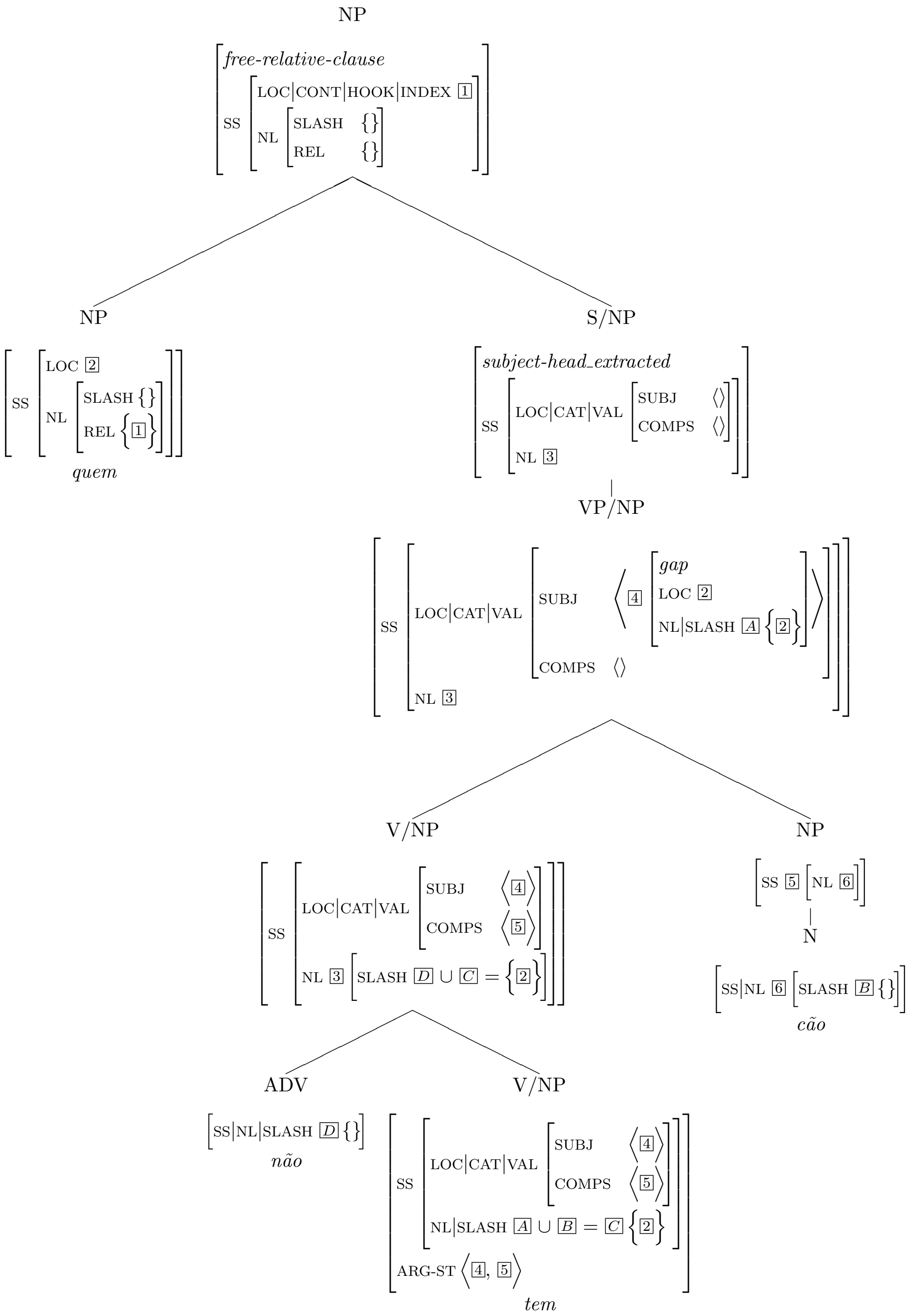


Figure 11.3: Example of a free relative clause. The clause is "quem não tem cão" (*who doesn't own a dog*).

The feature FREE is thus used to control rule application (*clause_relative* or *free-relative-clause*).

11.3 Topicalization

(127b), repeated below, shows an example sentence with a topicalized constituent, i.e. the PP "disso" (*about that*) occurs on the left edge of the sentence even though it is a complement of the embedded verb.

(130) [Desse incidente] prometo que falaremos - depois.
about that incident I promise that we will talk later
About that incident I promise we will talk later.

Topicalization is treated similarly to relative movement. The same rules to introduce gaps are employed. However, the filler phrase that is used is different, as the resulting node has properties that are different from relative clauses, and the filler daughter has different constraints as well. The phrase to combine a topicalized constituent with a sentence (containing a gap) is called *topic-head*.

In LXGram, topic-head constructions are not considered to produce complementizer phrases, because in Portuguese topics can follow complementizers and additionally, elements that modify verb-headed constituents can precede topics, but not complementizers. Instead, they give rise to sentences.

Contrary to wh-movement, fronted constituents are regular constituents that just happen to be fronted. As such, only the feature SLASH is relevant to account for them.

The syntax rule *topic-head* constrains the right daughter to have a non-empty SLASH. The LOCAL of the left daughter is unified with an element in the right daughter's SLASH. Furthermore, the right daughter must be a sentence (HEAD of the type *verb* and empty valence lists), and these features are unified with the corresponding features of the mother node.

Figure 11.5 shows the analysis for the sentence in (130). It is worth mentioning that the syntax rules for null subjects (in that example two instances are used) constrain the appropriate SUBJ element to be a synsem of the type *unexpressed-synsem*. This type is defined as having empty NON-LOCAL features.

11.4 Wh-Questions

LXGram contains an implementation of wh-questions that covers the following aspects:

- Different interrogative wh-elements
- Proclisis: wh-questions show proclisis even when they occur as main clauses
- Main vs. embedded wh-questions
- Wh-questions with "é que"
- Subject-verb inversion
- In-situ wh- constituents
- Multiple wh-elements
- Constraints on islands
- Pied-piping

These aspects are presented in some detail in the next section. The sections after that describe the implementation.

11.4.1 Description of the Data

Wh-words include the following:

- Determiners: *qual*, *quais*, *que* “which”, *quanto*, *quanta*, *quantos*, *quantas* “how many”
- Adverbs: *onde*, *aonde* “where, where to”, *porque*, *porquê* “why”, *como* “how”, *quando* “when”
- Pronouns: *quem* “who”, *(o) que*, *(o) quê* “what”

Some of these items come in pairs: one element of the pair must be used at the beginning of the sentence, the other cannot and often occurs at the very end of the sentence. This is the case of the pairs *o que*, *o quê* “what” and *porque*, *porquê* “why:”

(131) a. O que simboliza uma estrela?
what symbolizes a star?
What does a star symbolize?

b. * O quê simboliza uma estrela?
what symbolizes a star?

c. Uma estrela simboliza o quê?
a star symbolizes what?
What does a star symbolize?

d. * Uma estrela simboliza o que?
a star symbolizes what?

(132) a. Porque cintilam as estrelas?
why twinkle the stars
Why do stars twinkle?

b. * Porquê cintilam as estrelas?
why twinkle the stars

c. As estrelas cintilam porquê?
the stars twinkle why
Why do stars twinkle?

d. * As estrelas cintilam porque?
the stars twinkle why

In this text, we will call the first variant the *weak form*, because all its vowels are phonetically reduced (a phenomenon that affects unstressed vowels in Portuguese). We will call the second variant the *strong form*, because the last vowel (written with *ê*) does not generally occur in unstressed syllables in European Portuguese.

The strong element can in fact occur at the beginning of the sentence, but only if the sentence contains no finite verb. By contrast, the weak elements produce ungrammatical sentences in these cases:

(133) a. Porquê agora?
why now
Why now?

b. * Porque agora?
why now

(134) a. Porquê comportar-se assim?
why to behave so
Why behave like that?

b. * Porque comportar-se assim?
why to behave so

Multiple wh-consituents are possible in wh-questions (135a), but only one of them can appear in the left periphery of the sentence (135b):

(135) a. Quem dá boleia a quem?
who gives ride to who
Who gives who a ride?

b. * Quem a quem dá boleia?
who to who gives ride

In these questions with multiple wh-elements, all but the first one must occur in their strong form:

(136) a. O que leva a quê?
what leads to what
What leads to what?

b. * O que leva a que?
what leads to what

Wh- interrogative clauses can occur as the main (or matrix) clause (137a) or they can be embedded in another clause (137b):

(137) a. O que aconteceu ontem?
what happened yesterday
What happened yesterday?

b. Não sabemos o que aconteceu ontem.
not we know what happened yesterday
We don't know what happened yesterday.

In matrix wh-questions, the fronted wh-constituent is followed by either the verb, which is inverted with the subject (138a), or by the expression *é que* (138b). One of these two strategies (the use of *é que* or subject-verb inversion) is mandatory in non-echo questions (138c), but the two can also co-occur (138d):

(138) a. O que achas tu?
what think you
What do you think?

b. O que é que tu achas?
what you think
What do you think?

c. * O que tu achas?
what you think

d. O que é que achas tu?
what think you
What do you think?

(138c) is possible as an echo question. Because LXGram does not have access to discourse context (which is necessary to isolate echo questions), the implementation allows all four possibilities.

In embedded wh-questions all four possibilities are valid, although subject-verb inversion is sometimes strange:

(139) a. ? Sabes o que acho eu?
you know what think I
Do you know what I think?

b. Sabes o que é que eu acho?
you know what I think
Do you know what I think?

c. Sabes o que eu acho?
you know what I think
Do you know what I think?

d. ? Sabes o que é que acho eu?
you know what think I
Do you know what I think?

The expression *é que* literaly translates to English as *is it that*: *O que é que sabes?* literally means "What is it that you know?".

However, it cannot be analyzed as actually containing a verb form of *ser* "to be", because it does not show plural number agreement in contexts where that would be expected. The following examples show that wh-constituents introduced by *que* "which" can trigger plural agreement in the verb:

(140) a. Que estrela cintila?
which star twinkles
Which star twinkles?

b. Que estrelas cintilam?
which stars twinkle
Which stars twinkle?

One would expect *é que* to behave similarly, but this is not the case. The plural form corresponding to *é*, *são*, cannot occur (141b):

(141) a. Que estrela é que cintila?
which star twinkles
Which star twinkles?

b. * Que estrelas são que cintilam?
which stars twinkle
Which stars twinkle?

c. Que estrelas é que cintilam?
which stars twinkle
Which stars twinkle?

For this reason and because (i) no word can occur inside *é que* and (ii) no part of *é que* (*é* or *que*) can be separated from the other, the implementation assumes that *é que* is syntactically atomic.

In a Portuguese wh- interrogative clause, clitic pronouns always precede the verb—even if that clause is the main clause:

(142) a. Porque nos mentes?
why us you lie
Why do you lie to us?

b. * Porque mentes-nos?
why you lie us
Why do you lie to us?

Prepositions are obligatorily pied-piped:

(143) a. De que são feitas as estrelas?
of what are made the stars
What are stars made of?

b. As estrelas são feitas de quê?
the stars are made of what
What are stars made of?

c. * Que são feitas as estrelas de?
what are made the stars of

Other elements that can be pied-piped in relative clauses cannot be so in wh-questions:

(144) a. . . . um agradável jardim, no centro do qual surge um moderno conjunto
. . . a pleasant garden, in the center of which appears a modern set
escultórico.
sculpture-like
. . . a pleasent garden, in the middle of which one finds a set of modern sculptures

b. * No centro do que é que surge um conjunto escultórico?
in the center of what appears a set sculpture-like

As in other languages and other types of long-distance dependencies, there is no limit on the distance between the wh-expression and its canonical position. In all following examples the wh-expression at the beginning of the sentence is the complement of the rightmost verb in the sentence:

(145) a. De que é que as estrelas são feitas?
of what the stars are made
What are stars made of?

b. De que é que parece que as estrelas são feitas?
of what it seems that the stars are made
What does it seem that stars are made of?

c. De que é que dizem que parece que as estrelas são feitas?
of what they say that it seems that the stars are made
What do they say it seems stars are made of?

d. De que é que achas que dizem que parece que as estrelas são
of what you think they say that it seems that the stars are made
feitas?

What do you think they say it seems stars are made of?

Finally, wh-questions, like other long-distance dependencies, are well-known to exhibit the extraction island effects first identified by [Ross, 1967]. A few examples follow. Note that the in-situ variants are often possible:

(146) a. Chove sempre que aparece quem?
rains whenever appears who
It rains whenever who appears?

b. * Quem chove sempre que aparece?
who it rains whenever appears

(147) a. A final foi entre o Benfica e quem?
the final was between the Benfica and who
The final was between Benfica and who?

b. * Quem é que a final foi entre o Benfica e?
who the final was between the Benfica and

(148) a. Todos perguntavam porque cintila o quê?
all asked why twinkles what

b. * O que é que todos perguntavam porque cintila?
what all asked why twinkles

11.4.2 Analysis and Implementation

The implementation closely follows the pre-existing types in the Matrix meant to model long-distance dependencies in general and wh-questions specifically. In turn, the Matrix follows the standard HPSG analysis of long-distance dependencies. It is based on a NON-LOCAL feature of synsems, which contains three subfeatures: SLASH, QUE and REL. In theoretical HPSG these features are all set valued. In LKB implementations they take difference lists as their values:

$$\begin{bmatrix} \textit{synsem} & \\ \text{LOCAL} & \textit{local-min} \\ \text{NON-LOCAL} & \begin{bmatrix} \text{SLASH} & \textit{diff-list} \\ \text{QUE} & \textit{diff-list} \\ \text{REL} & \textit{diff-list} \end{bmatrix} \end{bmatrix}$$

For wh-questions only SLASH and QUE are relevant, since REL is used only with relative clauses.

The wh-words listed in Section 11.4.1 appear in two varieties in the lexicon: one entry for those words that must occur in the left periphery of the sentence (some of which are phonetically weak); another entry for in-situ wh-words (which are all phonetically strong).

The in-situ, or strong, versions of these elements are not analyzed as involving movement at all. Accordingly, their QUE list is empty.

The entries for the wh-elements that appear in the left periphery of the sentence have a QUE list with one element in it. Just like SLASH and REL, QUE is also amalgamated in lexical entries. That is, the lexical entries for words which take syntactics arguments (subject or complements) have as the value of their QUE feature the append of the QUE features of all their arguments. Words which do not take any syntactic arguments come in the lexicon with an empty QUE list. The lexical entries for wh-words are an exception to this generalization. They have their QUE feature constrained to take as value a list with one element in it.

The same gap-introducing rules as the ones used for relative clauses and topicalizations are also used with wh-questions. This means that phrases which contain a gap will have a non-empty

SLASH list, due to the fact that gap-introducing rules introduce an element with a non-empty SLASH and to SLASH amalgamation.

There is one special filler phrase type for wh-questions, *wh-question-phrase*, used in both matrix questions and embedded questions. This phrase type requires the left daughter (which is the filler daughter) to have a non-empty QUE difference list. Its right daughter must contain an element in SLASH (in other words, it must contain a gap), which is unified with the LOCAL feature of the left daughter (the gap must have syntactic constraints compatible to those of the filler daughter).

The expression *é que* is analyzed as a complementizer that takes as its complement a sentence with a non-empty SLASH. The right daughter of a *wh-question-phrase* is either a sentence with a non-empty SLASH or a complementizer phrase headed by *é que*.

The distribution of the weak vs. strong interrogative constituents is controlled in the following way. The mechanism that is set up for negative concord (see Chapter 13) is also used to force strong elements to occur after the verb in same clause. As mentioned in that Chapter, in each *sign* there is a feature which records its position relative to the verb occurring in the same clause. Namely, the boolean feature SYNSEM|LOCAL|CAT|POLARITY|EDGES|LEFT-EDGE-AFTER-VERB shows the positive value if and only if that *sign* occurs after that verb. The lexical entries for the elements that can only occur in-situ, viz. *(o) quê* "what" and *porquê* "why", constrain this feature appropriately (they require it to be of the type $+$).

Word order in wh-questions is not treated specially, because LXGram already accepts all but SOV word order in the general case: there is a subject-head syntax rule (together with head-complement rules, it produces SVO word order) as well as a head-subject syntax rule; the head-subject syntax rule is allowed to apply lower (producing VSO word order) or higher (VOS) than the head-complement rules. Verbal modifiers are allowed to modify all verb-headed phrases at either side. SOV word order is only allowed when the object is a clitic pronoun.[1]

Together with the filler rule for wh-questions (the *wh-question-phrase*), the syntactic configurations that normally produce VOS or VSO word orders also yield subject-verb inversion in wh-questions.

The implementation thus slightly overgenerates with respect to word order, as it allows OSV sentences such as (138c). However, these sentences are acceptable as echo questions.

Questions with multiple wh-elements are analyzed as involving a fronted wh-constituent and all other wh-elements occurring in their in-situ versions.

In LXGram, sentence force is encoded in a feature SF (for sentence force) under events. This feature has the value *question* for questions. The type *wh-question-phrase* constrains the event in the right daughter to have an SF feature with this value.

Embedded wh-questions can occur as the complement of any verb that selects for an interrogative clause as its complement (by allowing its complement to have an SF feature with the value *question*). This is the case of the verb *perguntar* "ask". Its complement may be an embedded

[1] With respect to general word order constraints, LXGram is slightly undergenerating. In Portuguese, SOV word order is also possible in idiomatic expressions (1) or when the object is a quantifier like *tudo* "everything" or *nada* "nothing" (2):

(1) Gato escaldado de água fria tem medo.
cat scalded of water cold has fear
A scalded cat is afraid of cold water.

(2) o olho que tudo vê
the eye that everything sees
the all-seeing eye.

yes-no question or an embedded wh-question.

11.5 Exclamatives

Exclamative sentences containing wh- movement have not been implemented yet.

11.6 "Tough" Adjectives

The NP in (149) illustrates a typical case of tough-movement.

(149) uma imagem_{i} difícil de esquecer $-_{\text{i}}$
an image difficult to forget
an image difficult to forget

The typical HPSG analysis for such an example is that the adjective selects for a complement with a non-empty SLASH. In this example, the complement of the verb is a gap. This gap is not realized, but there is a correference relation between it and the noun that the adjective modifies, as indicated by the subscripts in the example. If the adjective is used predicatively, this relation holds between the gap and the subject instead.

The syntactic constraints on this sort of adjectives are that they select for a slashed complement, but the element in the complement's SLASH is removed from the adjective's SLASH. The SLASH of the adjective is still the union of the SLASH of its dependents, but the element in the complement's SLASH is removed: the slashed element of the complement is not consumed via a filler phrase, but by the adjective itself.

Semantically, an adjective like "difícil" has a propositional argument that corresponds to its sentential complement, and the SARG of the noun modified by the adjective (or the subject, if the adjective occurs in a predicative position) is unified with the SARG of the element in the complement's SARG:

$$\begin{bmatrix}
\text{SYNSEM} & \begin{bmatrix} \text{LOCAL|CAT} & \begin{bmatrix} \text{HEAD|MARKER|SELECT|LOCAL|CONT|HOOK|SARG } \boxed{1} & \\ \text{VAL} & \begin{bmatrix} \text{SUBJ} & \left\langle \boxed{2} \right\rangle \\ \text{COMPS} & \left\langle \boxed{3} \begin{bmatrix}\text{NON-LOCAL|SLASH} \left\{ \begin{bmatrix}\text{CONT|HOOK|SARG } \boxed{1}\end{bmatrix}\right\}\end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix} \\ \text{NON-LOCAL } \boxed{4} & \end{bmatrix} \\
\text{ARG-ST} & \left\langle \boxed{2} \begin{bmatrix}\text{LOCAL|CONT|HOOK|SARG } \boxed{1} \\ \text{NON-LOCAL } \boxed{4}\end{bmatrix}, \boxed{3} \right\rangle
\end{bmatrix}$$

11.7 Islands

In this subsection, it is described how LXGram blocks extraction from islands, syntactic contexts from which nothing can be extracted (strong islands) or from which arguments can be extracted but adjuncts cannot (weak islands).

Strong islands:

- Clausal complements of nouns

(150) * a pessoa que há o boato [de que - desapareceu]
the person that there is the rumor that disappeared

Nouns with clausal complements have not been implemented yet, but this island can be easily accounted for by constraining the element in the COMPS list of these nouns to have an empty SLASH.

- Preverbal clausal subjects

(151) a. * as caixas que [que - fossem arrumadas] seria conveniente
the boxes that that were sorted out would be convenient

b. as caixas que seria conveniente [que - fossem arrumadas]
the boxes that would be convenient that they were sorted out
the boxes that it would be convenient to sort out

In LXGram, there are two head-subject constructions: one for subjects preceding the head, *subject-head*, and another for subjects following the head, *head-subject*. In the syntax rule *subject-head*, the non-head daughter is constrained to have an empty SLASH.

- Extraction from an adjunct

(152) a. o restaurante de que estamos $[_{\text{COMP}}$ mais perto -]
the restaurent of which we are more/most close
the restaurant we are closest to

b. * o restaurante de que tivemos o acidente $[_{\text{ADJ}}$ mais perto -]
the restaurant of which we had the accident more/most near

In LXGram, the rules that are responsible for adjuncts (see Section 5.3) constrain the non-head daughter to have an empty SLASH.

- Extraction from a conjunct

(153) * as caixas que [chovia] e [a Maria arrumava -]
the boxes that it rained and the Maria sorted out

LXGram implements the HPSG NON-LOCAL constraints on coordination. In particular, in the coordination rules, the SLASH of the mother is unified with the SLASH of all daughters. This constraint blocks examples like the one above, and allows for examples of Across-the-Board extraction like the following:

(154) as caixas que [o João comprou -] e [a Maria partiu -]
the boxes that the João bought and the Maria broke
the boxes that João bought and Maria broke

- Extraction of a conjunct

(155) * as caixas que o João arrumou [-] e aqueles livros
the boxes that the João sorted out and those books

(156) * as caixas que o João arrumou aqueles livros e [-]
the boxes that the João sorted out those books and

In LXGram gaps are introduced via the rules presented in Section 11.1: there are rules to introduce gaps in subject, complement and adjunct positions. Coordination is treated with a set of rules separate from the rules for subject, complements and adjunts (see Chapter 10). As such, there is no way to introduce a gap as a conjunct.

Weak islands:

- Relative clauses

(157) a. a pessoa a quem nunca sabes [o que deves dizer -]
the person to whom never you know what you should say
the person who you never know what to say to

b. * o cuidado com que nunca sabes [o que deves dizer -]
the caution with which never you know what you should say

For efficiency reasons, SLASH is constrained to have at most one element. This constraint blocks the required analyses for the syntactic constituents "dizer" and "deves dizer" in both examples, which would have two elements in its SLASH, one for the direct object and another one for the indirect object or adjunct.

- Embedded interrogative clauses

(158) a. a pessoa que não sabemos [quando - volta]
the person that not we know when returns
the person such that we don't know when he returns

b. * a altura em que não sabemos [quem volta -]
the time when not we know who returns

LXGram does not support interrogatives yet, but examples like these would also require a SLASH with two elements in the embedded clause.

- Postverbal clausal subjects

(159) a. as caixas que seria conveniente [que arrumássemos -]
the boxes that would be convenient that we sorted out
the boxes which it would be convenient for us to sort out

b. ?? o local onde seria conveniente [que tivéssemos arrumado as caixas -]
the place where would be convenient that we had sorted out the boxes

As mentioned above, the syntax rule for preverbal subjects constrains the subject daughter to have an empty SLASH, but the syntax rule for postverbal subjects does not. Both of these examples are accepted by LXGram.

- Complements of negative verbs and complements of factive verbs verbs[2]

[2] Factive verbs are verbs that select for a propositional complement and entail its truth.

(160) a. o local onde negam [que estavam -]
the place where they deny that they were
they place where they deny that they were

b. * a altura em que negam [que tiveram um acidente -] ?
the time when they deny that they had an accident

(161) a. o local onde lamentam [que estivessem -]
the place where they regret that they were
the place where they regret that they were

b. * a altura em que lamentam [que tenham tido o acidente -]
the time when they regret that they had the accident

Currently, negative verbs, like "negar" (*deny*), and factive verbs, like "lamentar" (*regret*), are not distinguished in the grammar's lexicon from other verbs with a sentential complement, like "dizer" (*say*), which allow adjunct extraction from their complement:

(162) a altura em que dizem [que tiveram o acidente -]
the time when they say that they had the accident
the time when they say they had the accident

Therefore, this phenomenon is currently unaccounted for.

11.8 Postponed Coverage or Known Limitations

The distribution of some relative elements with pied-piped constituents is currently not properly accounted for: cf. "*o preço de que" vs "o preço dos quais" (*the price of which*). The relative pronoun "que" can appear as a filler on its own or with a pied-piped preposition, but it cannot appear in a filler that contains a noun. The current implementation overgenerates in that regard.

Some long distance phenomena have not been implemented yet. The most important are wh-questions (matrix or embedded).

$$\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{x2}\ x \\
\text{RELS} & \left\langle
\begin{bmatrix} \textit{free-relative\_q\_rel}\langle 0{:}4\rangle & \\ \text{LBL} & \boxed{h1} \\ \text{ARG0} & \boxed{x2} \\ \text{RSTR} & \boxed{h3}\ h \\ \text{BODY} & \boxed{h4}\ h \end{bmatrix},
\begin{bmatrix} \textit{person\_n\_rel}\langle 0{:}4\rangle & \\ \text{LBL} & \boxed{h5}\ h \\ \text{ARG0} & \boxed{x2} \end{bmatrix},
\begin{bmatrix} \textit{neg\_rel}\langle 5{:}8\rangle & \\ \text{LBL} & \boxed{h5} \\ \text{ARG0} & \boxed{e6}\ e \\ \text{ARG1} & \boxed{h7}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_ter\_v\_rel}\langle 9{:}12\rangle & \\ \text{LBL} & \boxed{h8}\ h \\ \text{ARG0} & \boxed{e9}\ e \\ \text{ARG1} & \boxed{x2} \\ \text{ARG2} & \boxed{x10}\ x \end{bmatrix},
\begin{bmatrix} \textit{udef\_q\_rel}\langle 13{:}16\rangle & \\ \text{LBL} & \boxed{h11}\ h \\ \text{ARG0} & \boxed{x10} \\ \text{RSTR} & \boxed{h12}\ h \\ \text{BODY} & \boxed{h13}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_cão\_n\_rel}\langle 13{:}16\rangle & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{x10} \end{bmatrix}
\right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h3} \\ \text{LARG} & \boxed{h5} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h7} \\ \text{LARG} & \boxed{h8} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h12} \\ \text{LARG} & \boxed{h14} \end{bmatrix}
\right\rangle
\end{bmatrix}$$

Figure 11.4: Example MRS representation of a free relative clause. The clause is “quem não tem cão” (*who doesn’t own a dog*).

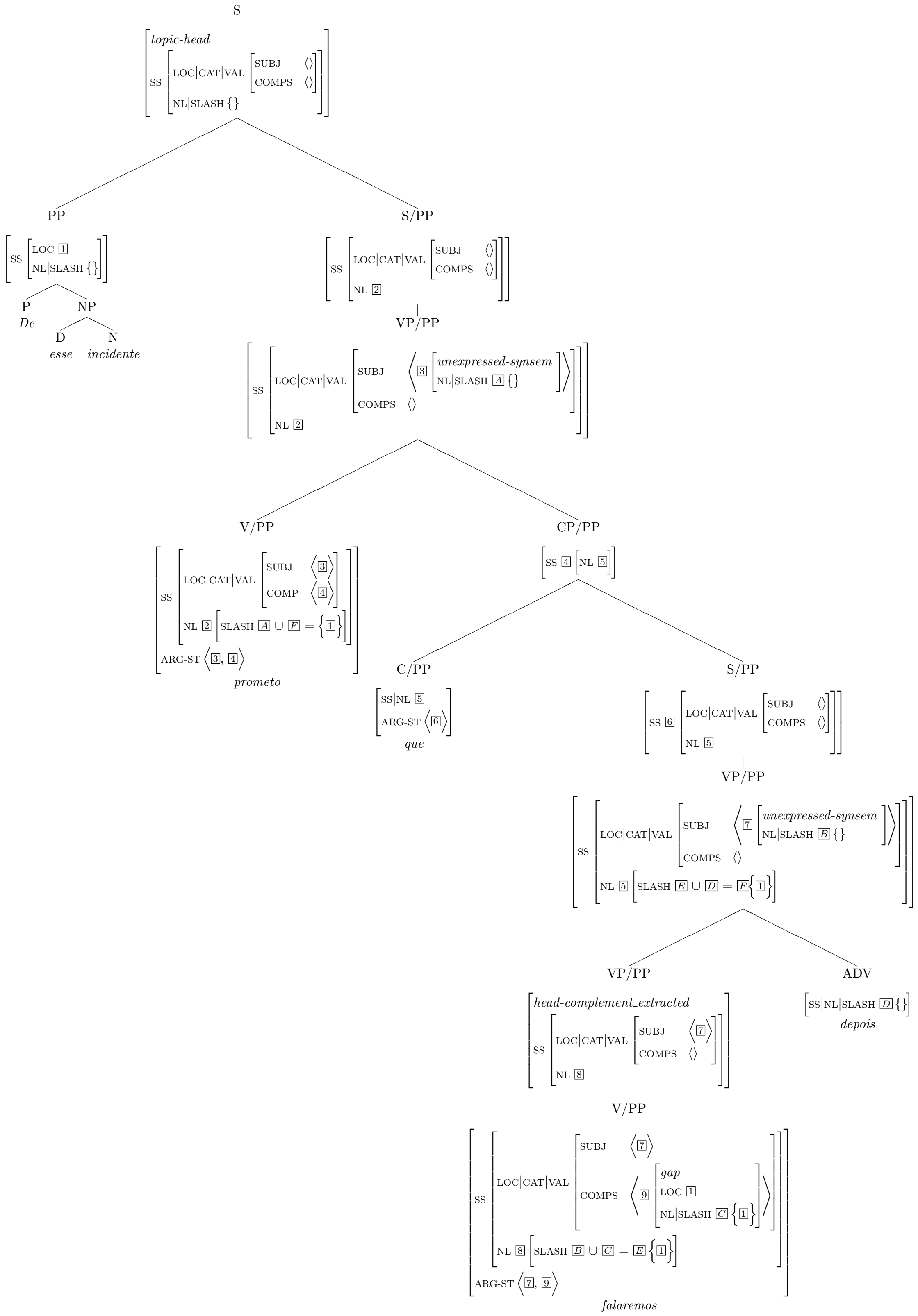

S
topic-head
SS LOC|CAT|VAL SUBJ ⟨⟩ COMPS ⟨⟩
NL|SLASH {}
PP
SS LOC 1 NL|SLASH {}
P De
NP
D esse
N incidente
S/PP
SS LOC|CAT|VAL SUBJ ⟨⟩ COMPS ⟨⟩
NL 2
VP/PP
SS LOC|CAT|VAL SUBJ ⟨3 [unexpressed-synsem NL|SLASH A {}]⟩ COMPS ⟨⟩
NL 2
V/PP
SS LOC|CAT|VAL SUBJ ⟨3⟩ COMP ⟨4⟩
NL 2 [SLASH A ∪ F = {1}]
ARG-ST ⟨3, 4⟩
prometo
CP/PP
SS 4 [NL 5]
C/PP
SS|NL 5
ARG-ST ⟨6⟩
que
S/PP
SS 6 LOC|CAT|VAL SUBJ ⟨⟩ COMPS ⟨⟩
NL 5
VP/PP
SS LOC|CAT|VAL SUBJ ⟨7 [unexpressed-synsem NL|SLASH B {}]⟩ COMPS ⟨⟩
NL 5 [SLASH E ∪ D = F{1}]
VP/PP
head-complement_extracted
SS LOC|CAT|VAL SUBJ ⟨7⟩ COMPS ⟨⟩
NL 8
ADV
SS|NL|SLASH D {}
depois
V/PP
SS LOC|CAT|VAL SUBJ ⟨7⟩ COMPS ⟨9 [gap LOC 1 NL|SLASH C {1}]⟩
NL 8 [SLASH B ∪ C = E {1}]
ARG-ST ⟨7, 9⟩
falaremos

Figure 11.5: Analysis for a sentence with a topicalized PP. The sentence is "Desse incidente prometo que falaremos depois" (*About that incident I promise we will talk later*).

Chapter 12

Tense and Aspect

LXGram contains an implementation of tense and aspect that assigns to the various Portuguese verb tenses meaning representations inspired by the literature on the semantics of tense and aspect.

The various tenses are described through temporal relations between special times: the speech time (or utterance time), which refers to the moment when the sentence was uttered; the event time, which refers to the time when the situation described by the verb holds; a reference time, which often mediates the temporal relation between the other two. This is inspired on the work of [Reichenbach, 1947], but takes into account refinements introduced in later work.

Another aspect of the implementation of tense and aspect in LXGram is an encoding of aspectual constraints in the semantic representations. This is inspired by the work of [de Swart, 1998a; de Swart, 2000; Bonami, 2002].

12.1 Aspectual Type

Aspect and aspectual type are related to the way situations are described in natural language with respect to their internal structure ([Vendler, 1957; Comrie, 1976; Moens, 1987; Binnick, 1991; Verkuyl, 1993; Smith, 1997]).

[Vendler, 1957; Vendler, 1967] notes that not all verbs behave identically as far as grammatical tense is concerned. For instance, *I am running* is a valid English sentence, but *I am knowing* is nonsense. This phenomenon is now known as aspectual type, aspectual class, situation type, Aktionsart, lexical aspect, Vendler class or Vendler/Dowty class.

Vendler introduced four aspectual classes: states, activities, accomplishments and achievements. In this text we will use the terminology of [Dowty, 1979], though, and talk about states, processes (Vendler's activities), culminated processes (Vendler's accomplishments), and culminations (Vendler's achievements).[1]

Examples of states are *to hate beer*, *to know the answer*, *to own a car*, *to stink. to be sick.* Examples of processes are *to work*, *to eat ice cream*, *to grow*, *to play the piano.* Among culminated processes we find *to paint a picture*, *to burn down*, *to deliver a sermon.* Finally the class of culminations contains phrases such as *to explode*, *to win the game*, *to find the key.*

States and processes are atelic situations in that they do not make salient a specific instant in time. Culminated processes and culminations are telic situations: they have an intrinsic, instantaneous endpoint, called the culmination (e.g. in the case of *to paint a picture*, it is the moment when the picture is ready; in the case of *to explode*, it is the moment of the explosion).

[1] In some of the literature, culminations are further divided into culminations, *stricto sensu*, and points, but we will ignore this distinction.

Culminated processes consist of a process followed by a culmination (e.g. *to paint a picture* is a process of painting a picture and a culmination of finishing it).

These classes are distinguished by several linguistic tests. One such test is their occurrence in the progressive: processes and culminated processes have no problem appearing in the progressive (*He is running*, *He is painting a picture*), whereas states and culminations often produce ungrammatical sentences (** He is knowing French*, ** He is recognizing his friend*). Another test is the preposition used in durational adverbials: the duration of processes is indicated by durational phrases headed by *for* (*John swam for two hours*), whereas *in* is used with culminated processes to indicate the duration of the process that precedes the culmination (*John painted a picture in two hours*).

Aspectual type is not a property of words, but rather of phrases. Different phrases with the same head verb can have different aspectual types. For instance *to paint a picture* is a culminated process (cf. *John painted a picture in two hours*), but *to paint pictures* is a process (cf. *John painted pictures for two hours*) ([Garey, 1957; Verkuyl, 1972; Platzack, 1979; Krifka, 1992]). Additionally, some phrases have an aspectual type different from the aspectual type of their composing elements: *to paint a picture* is a culminated process but *to paint a picture every day* is a process (cf. *John painted a picture every day for two years*). In this example, the phrase *every day* combines with a phrase that describes a culminated process to produce a larger phrase that describes a process. This is known as aspect shift (or *Aktionsart* shift).

A phenomenon related to aspect shift is aspect coercion: clashes of constraints on aspectual type often do not result in ungrammatical expressions but rather force a coercion of their aspectual type, with a noticeable shift in their meaning. For instance, *for* adverbials, as mentioned above, combine with processes. However, a sentence like *John painted a picture for two hours* is grammatical, but the culminated process *to paint a picture* is coerced into a process, with a change in meaning: the sentence no longer means that John finished the painting (the culmination is stripped as the result of the coercion).

Aspectual coercion provides an explanation for the progressive/imperfective paradox ([Bach, 1986; Dowty, 1979]), illustrated by the examples in (163) and (164).

(163) a. John was swimming.
b. John swam.

(164) a. John was painting a picture.
b. John painted a picture.

The paradox is that (163a) entails (163b), but (164a) does not entail (164b). This contrast is due to both (163a) and (163b) describing atelic situations, but (164b) contains a culminated process (the picture was finished), whereas (164a) is an atelic situation (the picture was not finished). The idea is that the progressive construction combines with processes, which are atelic. In (164a), the progressive construction coerces the culminated process of *painting a picture* into a (non-culminated) process.

The work of [de Swart, 1998b; de Swart, 2000] analyzes aspectual coercion as the occurrence of implicit aspectual operators that are used only when clashes occur. Just like the progressive is an aspectual operator, namely a function from processes to states, there are other aspectual operators that are different in that they are silent. The sentence in (165), together with a schematic representation of the relative scope between the different temporal and aspectual elements involved and taken from [de Swart, 1998b], illustrates this idea of implicit aspectual operators. In this case the silent operator is represented with C_{eh}, and it is a function from events (the author reserves this term to refer to telic situations; *John played the sonata* is a telic situation) to

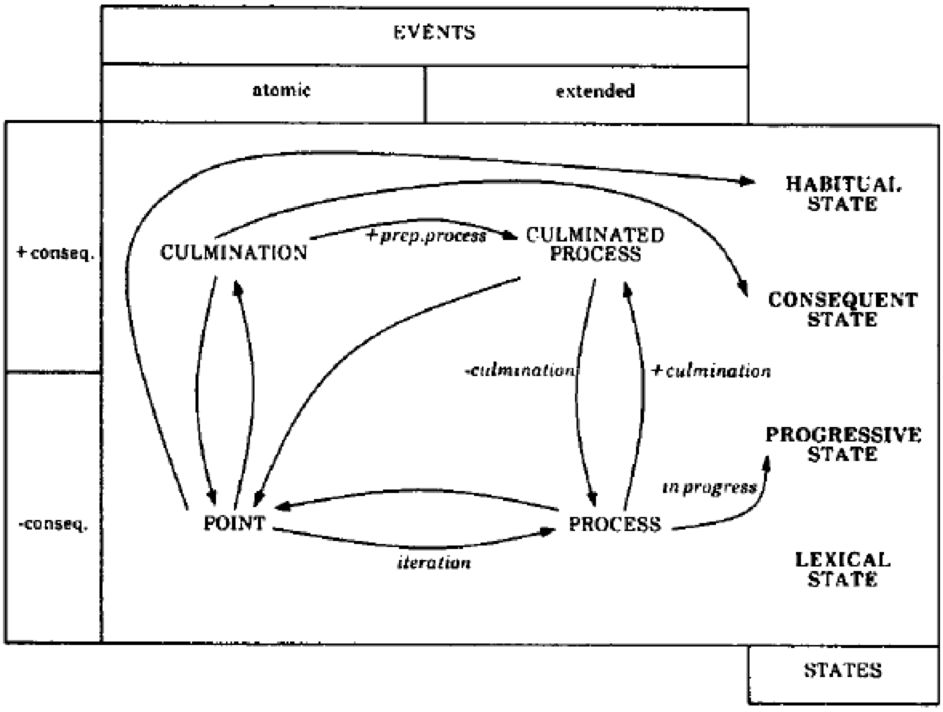


Figure 12.1: Possible kinds of aspectual type coercion according to Moens and Steedman

homogeneous (i.e. atelic) situations (as required by the *for* adverbial, as mentioned above).[2]

(165) John played the sonata for eight hours.
[PAST [FOR eight hours [C_{eh} [John play the sonata]]]]

[Moens and Steedman, 1988] introduce the concept of "event *nucleus*": an event has a nucleus made of a preparatory process followed by a culmination followed by a consequent state. The examples in (166) mention an event of building a bridge:

(166) a. *When they built the 59th bridge, they used the best materials.*
b. *When they built the 59th bridge, they solved most of their traffic problems.*

The preparatory process is the process of actively building the bridge, the culmination is the point when bridge is finished, and the consequent state is the existence of that bridge.

In these examples, the *when* clause can refer to different parts of the nucleus of *building a bridge*. The *when* clause refers to the preparatory process in (166a), and to the consequent state in (166b).

According to [Moens and Steedman, 1988], the components of this nucleus are optional, and their presence or absence is what determines aspectual type. Aspectual coercion can thus be viewed as adding or removing parts of the nucleus. The authors introduced a diagram describing the possible transitions involved in aspectual type coercion, which we show in Figure 12.1.

[Pustejovsky, 1991] explains aspectual phenomena by viewing situations as structures composed of other situations. For instance, the situations described in (167) are analyzed as having an internal structure. More specifically and as depicted below in (168), each of the two sentences

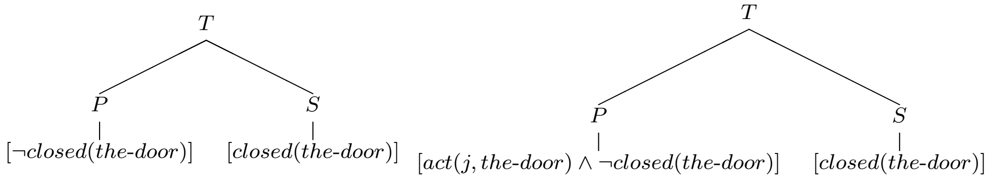
[2] Atelic situations are called homogeneous because they exhibit the subinterval property: if they hold in some time interval t, they hold in every subinterval of t.

is viewed as describing a transition T between a first situation when the door is not closed (P) and a second situation when the door is closed (S).

(167) a. The door closed.

b. John closed the door.

(168)



(167a) is a culmination and (167b) is a culminated process. For [Pustejovsky, 1991], the difference between culminations and culminated processes is that the P part of the latter also includes an *act*(*ivity*) predicate (as seen in (168)) between the two participants of the situation and this activity causes the change of state (the transition from [¬*closed*(*the-door*)] to [*closed*(*the-door*)]). Such a representation captures the fact that some phrases can modify parts of the situations described in sentences. For instance, *almost* is ambiguous with culminated processes. A sentence like *John almost closed the door* can mean that John never started the process of closing it or that he did but he did not finish it. In the second interpretation *almost* scopes only over the S structure in the representation above.

Aspectual type has several consequences for the way in which the meaning of sentences can be computed, i.e. compositional semantics. Discourse Representation Theory (or DRT; [Kamp and Reyle, 1993]) is one of the most influential current theories of compositional semantics. It assumes a representation of tense inspired by the work of [Reichenbach, 1947], describing tense with the help of several points in time. DRT features different modes of composing meaning representations in the presence of temporal location adverbials (e.g. *yesterday*, *last week*, *in 1974*, etc.), depending on the aspectual type of the verb. For states, it assumes that the time in which the state is true overlaps the time picked up by these expressions (cf. *John was ill yesterday*). In the case of non-stative situations, this relation is more specifically one of inclusion (cf. *John broke his ankle yesterday*).

12.2 The Semantic Representation of Tense

The following running example illustrates the implementation of tense and aspect:

(169) *A atriz mudou-se de França para os Estados Unidos em fevereiro de 1947.*
the actress moved from France to the United States in February of 1947
The actress moved from France to the United States in February 1947.

The MRS representation for this sentence, as produced by the grammar, is shown in Figure 12.2.

It is important to distinguish between grammatical tense and semantic tense: we will use the first expression to refer to inflectional morphology alone, and the second one to refer to the temporal and aspectual meanings conveyed by grammatical tenses.

Each predicate denoted by a verb, adjective, preposition or adverb receives a Davidsonian semantic representation ([Davidson, 1967; Parsons, 1990]), with an event variable as its first argument. This variable is not explicitly quantified, but assumed to be bound by an existential quantifier. This is in line with a substantial amount of the HPSG literature, including computational implementations such as the English Resource Grammar ([Baldwin et al., 2005]) and the Grammar Matrix ([Bender et al., 2002]). An example is the predicate *_mudar_v* (for the

$<h1,$
$\{h3\colon \_o\_q(x4, h5, h6),$
$h7\colon \_atriz\_n(x4),$
$h8\colon at(e2\ \{culmination\colon\ +\}, t9),$
$h8\colon before(t9, t10\ \{t\text{-}value:\ utterance\text{-}time\}),$
$h8\colon aspectual\text{-}operator(e2, e12, h11),$
$h11\colon \_mudar\_v(e12, x4),$
$h11\colon \_de\_p(e14, e12, x13),$
$h15\colon proper\_q(x13, h16, h17),$
$h18\colon named(x13,\ \text{"França"}),$
$h11\colon \_para\_p(e20, e12, x19),$
$h21\colon \_o\_q(x19, h23, h22),$
$h24\colon named(x19,\ \text{"Estados Unidos"}),$
$h11\colon \_em\_p(e26, e12, x25),$
$h27\colon udef\_q(x25, h28, h29),$
$h30\colon \_fevereiro\_n(x25),$
$h30\colon \_de\_p(e31, x25, x32),$
$h33\colon proper\_q(x32, h34, h35),$
$h36\colon named(x32,\ \text{"1947"})\},$
$\{h1 =_q h8,\ h5 =_q h7,\ h16 =_q h18,\ h23 =_q h24,\ h28 =_q h30,$
$h34 =_q h36\} >$

Figure 12.2: MRS for *A atriz mudou-se de França para os Estados Unidos em fevereiro de 1947* "The actress moved from France to the United States in February 1947"

verb form corresponding to English "move") in Figure 12.2: its first argument ($e12$) is an event variable.

Additionally, an *at* relation is employed that relates this event variable with a temporal index (in Figure 12.2 this relation is labeled with $h18$). Temporal indices refer to times. In the existing literature on tense, some authors use quantified time variables, while other authors use free time variables. [Partee, 1973] presents arguments for a free variable approach. Our temporal indices are compatible with this approach. This temporal index occurring as an argument of the *at* relation represents the event time. Temporal indices appear with the type t in the MRS representations, and a feature T-VALUE is appropriate for them. This feature locates the index in the time line.

Depending on the grammatical tense, there are then temporal relations between temporal indices, in the spirit of Reichenbach, who describes tense as temporal relations between three times: the event time E, the reference time R and the speech time or utterance time S. For our purposes, we do not need full Reichenbachian representations for many of the tenses: in some cases we will represent the temporal relation between the event time and the speech time directly, and say nothing about the reference time. For instance, we assume semantic present to be a temporal relation between S and E, in particular a temporal overlap relation. We follow Discourse Representation Theory ([Kamp and Reyle, 1993, p. 541]) in further assuming that the speech time is seen as punctual, which means that this overlap relation is more specific than just overlap, and it is an inclusion relation: the event time includes the utterance time.

In our example sentence in (169), the Portuguese verb is in the *pretérito perfeito* tense. The semantics of this tense is ambiguous between a simple perfective past (i.e. the situation occurred in the past and is culminated; in Reichenbach's system, E and R are simultaneous and R precedes

S) and a present perfect (the situation has a resulting state that holds and is relevant at the present; in Reichenbach's system, E precedes R and R and S are simultaneous). Since it is not possible to underspecify this distinction in the semantic representations, there are two options: duplicate the number of analyses provided by the grammar for each verb with this tense in the input (this is the approach of [Van Eynde, 2000], for Dutch, but it is computationally costly and does not seem justifiable as both representations essentially describe a past event); or use a simplified representation that covers both interpretations. We chose the second route. The event time is before the utterance time and, accordingly, there is a temporal relation *before* with the event time as its first argument in the MRS representations.

The second argument of the temporal relation $before$ is another temporal index with a T-VALUE specified to have the value *utterance-time*. This is how the speech time is represented. According to what has been presented so far, the relevant representation fragment is thus:

$$at(e2,\ t9)\ \wedge\ before(t9,\ t10\ \{\textit{t-value}:\ \textit{utterance-time}\})\ \wedge \\ \_mudar\_v(e2,\ x4)\ \ldots$$

That is, the event described by the form of the verb *mudar* "move" occurred in a time that precedes the utterance time. In this text, we will sometimes use s to represent the speech time, as short-hand notation for a temporal index with the value *utterance-time* for its T-VALUE feature, as in:

$$at(e2,\ t9)\ \wedge\ before(t9,\ s)\ \wedge\ \_mudar\_v(e2,\ x4)\ \ldots$$

Grammatical tense presents two levels of ambiguity that must be resolved:

- The same form can correspond to more than one grammatical tense. An English example is the verb form *put*, which can, for instance, be present tense or past tense. Portuguese also contains similar ambiguities, e.g. forms like *corremos* ("we run" or "we ran").
- The same grammatical tense can cover more than one meaning when it comes to locating a situation in time. An English sentence like *I leave tomorrow* shows that present tense can refer to the future. Usually this tense locates an event in the present. Portuguese shows similar cases.

This two-fold ambiguity is accounted for by a two-layer analysis in the working grammar. The first layer consists in a set of rules that map surface form to grammatical tense. The second layer consists in a set of rules that map grammatical tense to semantic representations of tense. Both are implemented as lexical rules, i.e. unary rules that apply to single lexical items (verb forms in this case).

We distinguish between imperfective and perfective tenses as they occur in Portuguese as well as several other languages (e.g. the remaining Romance languages or Slavic languages[3] or Greek). This distinction interacts with aspectual type (our encoding of aspect in MRS representations is presented in the next section): perfective tenses constrain the whole clause to be telic whereas imperfective tenses constrain it to be atelic ([de Swart, 1998a; de Swart, 2000; Bonami, 2002; Flouraki, 2006]).

We assume that present cannot be perfective and, similarly to [Michaelis, 2011], that languages without perfective vs. imperfective distinctions show ambiguity in the other tenses. The examples in (170) are hers and support this last claim.

[3] Slavic languages are usually analyzed not as having perfectivity distinctions in their tense system but rather as having perfective and imperfective verbs (i.e. most verbs come in pairs formed by a perfective verb and an imperfective verb). This distinction is irrelevant for this discussion.

Semantic imperfective present:	"...fuma"	("...smokes")
$at(e,t) \wedge includes(t,s) \wedge \ \_fumar\_v(e,x)\ldots$		
Semantic imperfective past:	"...fumava"	("...smoked")
$at(e,t1) \wedge overlap(t1,t2) \wedge before(t2,s) \wedge \ \_fumar\_v(e,x)\ldots$		
Semantic perfective past:	"...fumou"	("...smoked")
$at(e,t) \wedge before(t,s) \wedge \ \_fumar\_v(e,x)\ldots$		

Table 12.1: The meaning of some tenses

(170) a. At the time of the Second Vatican Council, they *recited the mass* in Latin.

b. He lied to me and I *believed* him.

The highlighted phrase in the English sentence in (170a) is telic (cf. *They recited the mass* ***in*** *20 minutes*), but the sentence can nevertheless have an atelic reading (i.e., *...they used to recite the mass...*). In (170b) the highlighted verb is lexically stative, but the clause where it occurs can have a telic reading (i.e., *...I believed what he said at that one time*). This sort of aspectual coercion is similar to the one found with the perfective and imperfective past tenses in languages where the difference between them is marked. Therefore, the English simple past must be ambiguous between a perfective and an imperfective past tense.

Similarly, future tense (or future constructions) is ambiguous in English as well as Romance languages with respect to perfectivity, in contrast to languages like Russian, that shows perfectivity distinctions also in the future tenses.

The examples in Table 12.1 show the sort of temporal representation that we have in mind, using the verb *fumar* "smoke" for illustration. We leave future tense aside, as it adds nothing new to the discussion, although future tenses and future constructions are implemented in LXGram. As can be seen in this table, from the semantics for the imperfective past it does not follow that the event time does not overlap the speech time. This is indeed a possibility, and it contrasts with the perfective past. As the pair of sentences in (171) shows, the imperfective past can describe situations that still hold at present; similar sentences with the perfective past sound strange.

(171) a. O João ontem **estava** doente e hoje ainda está. (imperfective past)
John was ill yesterday and still is today.

b. ?? O João ontem **esteve** doente e hoje ainda está. (perfective past)
John was ill yesterday and still is today.

These representations are inspired by [Kamp and Reyle, 1993] and [Van Eynde, 1998]. They also make use of several times: the event time (identical to Reichenbach's E), the location time (similar to R, but in their work it is the time described by temporal expressions modifying the verb) and the perspective point (similar to S), and assume an interaction between the meaning of tenses and the aspectual type of the verb. In the case of the past tenses, these authors assume that the relation between the location time of a situation and the perspective point is determined by aspectual class. For states this is one of overlap. For non-stative situations this is, more specifically, one of temporal inclusion. It follows from the event time being included in the location time and the location time preceding the utterance time (the past tense semantics) that the event time also precedes the utterance time. This is essentially the simplified representation that we use here for the perfective past. Unlike these pieces of work, we do not make this distinction depend on the aspectual type of the verb but rather assume that it is the difference between imperfective and perfective tenses. It just happens that perfective tenses constrain the whole clause to be telic whereas imperfective tenses constrain it to be atelic [de Swart, 1998a;

de Swart, 2000; Bonami, 2002; Flouraki, 2006], which means that imperfective tenses trigger no aspect shift when they combine with states, and neither do perfective tenses when they combine with culminations or culminated processes. The following Portuguese examples, based on those in (170) above, motivate our departure from their analysis:

(172) a. Na altura do Concílio Vaticano II, recitaram a missa em Latim. (perfective)
At the time of the Second Vatican Council, they recited the mass in Latin (they did that just once).

b. Na altura do Concílio Vaticano II, recitavam a missa em Latim. (imperfective)
At the time of the Second Vatican Council, they recited the mass in Latin (they used to do that).

(173) a. Ontem acreditei nele. (perfective)
Yesterday I believed him (I believed what he said yesterday).

b. Ontem acreditava nele. (imperfective)
Yesterday I believed him (Yesterday I was under the assumption that he always speaks the truth).

The examples in (172) both exhibit the phrase *recitar a missa* "recite the mass", which is a culminated process (i.e. a telic situation). The sentences in (173) contain the stative verb *acreditar* "believe". In all cases there is a preposition phrase or an adverb that locates the described situations in time (i.e. this temporal expression identifies the location time). The examples with the perfective forms describe situations that happen only once and within the time interval referred to by these modifiers. The imperfective sentences describe situations that are more prolonged in time and may extend outside the boundaries of these intervals.

12.3 The Semantic Representation of Aspect

Aspectual type is described for with the help of three Boolean features: CULMINATION (positive for culminations and culminated processes), PROCESS (positive for processes and culminated processes) and STATE (positive for states). These features are appropriate for event variables.

Aspectual type is in part a lexical property. It is known that it is difficult to annotate it ([Pustejovsky et al., 2006]). For this reason, in our implementation, we abstain from encoding aspectual type in the lexicon.

However, contextual (i.e. syntactic) constraints on aspect are indeed implemented. These are represented by aspectual operators, which are functions from situation descriptions to situation descriptions, and they appear as relations (i.e. elementary predications) in the MRS representations.

For instance, we represent a function from state descriptions to culmination descriptions as $\textit{aspectual-operator}(e_2\{culmination : +\}, e_1\{state : +\}, X)$. Here, e_1 is a state, e_2 is a culmination, and X is the semantic representation for the state e_1. The event variable (e_2 in this example) of the resulting situation is included in the representation, since Davidsonian representations are being used. We also make use of an extra argument, which is just a pointer for the event variable of the argument (e_1 in this example), because it can be useful when using MRS representations in applications.

We follow [Bonami, 2002] in assuming that all aspectually sensitive relations allow for at most one implicit aspectual operator. These implicit aspectual operators account for aspectual coercion. Therefore every context that allows aspectual coercion must introduce either zero or one aspectual operators in the semantic representation: zero if no aspectual coercion actually occurs, and one otherwise.

Because it is not possible to underspecify the number of relations in an MRS, one *aspectual-operator* is introduced in every aspectually sensitive context, although in general it is not specified which operator it is. That is, one underspecified operator is always introduced: this is the *aspectual-operator* predicate just mentioned. We assume that sometimes it stands for a dummy relation (i.e. the identity function), in the cases when no aspectual shift actually occurs.

Several elements are sensitive to aspectual type. Tense is one of them. Consider the two example sentences below. They correspond to the English sentence *Samuel had a son yesterday.*

(174) a. O Samuel teve um filho ontem.
b. O Samuel tinha um filho ontem.

The difference between the two is grammatical tense, but they also convey different temporal and aspectual meanings. In the first one the verb is in the *pretérito perfeito* (the perfective past). In the second one the verb is in the *pretérito imperfeito* (the imperfective past).

As already mentioned above, perfective aspect constrains the whole event to be telic (a culmination or a culminated process). Imperfective aspect constrains it to be a state in Portuguese. The first sentence means one of Samuel's sons was born yesterday, whereas the second one simply says that one of his sons existed yesterday.

The grammar assigns to the first sentence a semantic representation expressing this:

> $at(e\{culmination : +\},\ t)\ \wedge\ before(t,\ t2\{t\text{-}value:\ utterance\text{-}time\})\ \wedge$ $aspectual\text{-}operator(e,\ e2,\ ter(e2,\ X))$, where X is the representation for the verb's arguments.

This representation is similar to the one presented above in the discussion about tense, but it includes information about aspect as well. In particular, an *aspectual-operator* was added scoping over the relation for the main verb in this sentence. This operator is introduced in the semantics by the lexical rule responsible for semantic tense (together with the temporal relations seen in this MRS fragment), as tenses impose aspectual constraints at the clausal level ([Bonami, 2002]). The constraint that the event variable e is telic (its feature CULMINATION has the value +) also comes from the *pretérito perfeito* tense.

By contrast, the second sentence receives a representation like:

> $at(e\{state : +\},\ t)\ \wedge\ overlaps(t,\ t2)\ \wedge$
> $before(t2,\ t3\{t\text{-}value:\ utterance\text{-}time\})\ \wedge$
> $aspectual\text{-}operator(e,\ e2,\ ter(e2,\ X))$, where X is the representation for the verb's arguments.

Unlike the *pretérito perfeito* tense, which introduces an aspectual operator that produces telic situations, the *pretérito imperfeito* constrains the whole clause to be a state. In this example, this is encoded in the event variable e, with its feature STATE constrained to have the + value.

The verb *ter* "have", instantiating the third argument of the *aspectual-operator* relation, is a state. Even though lexical aspect is not encoded in the grammar (and therefore there is no restriction on the aspectual features of $e2$) for the reasons mentioned above, our encoding of aspect at the syntactic level, as it was just illustrated, is important because it can capture distinctions such as the one illustrated by this pair of sentences.

Additionally, it can be straightforwardly extended with lexical aspect: if we knew that "have" is lexically a state, then the *aspectual-operator* in the second sentence is a function from states to states (i.e. it is the identity function, and does not change the basic meaning of the verb). The aspectual operator in the first sentence would be a function from states to telic situations.

One such operator can be the inchoative operator, which is the correct reading for this sentence (i.e. the state begins to hold yesterday). This final step is not deterministic: for instance, in the example in (173a), we also find a coercion of a state into a telic situation caused by the perfective past, but in that example the result is not an inchoative interpretation, but a different kind of change in meaning. For this reason, we can not identify the exact aspectual operator in context, and we use this abstract *aspectual-operator* relation every time in the representations. This *aspectual-operator* can be seen as the supertype of all aspectual operators. The analysis of aspect coercion implemented in the grammar is essentially the same as the analysis of [de Swart, 1998a], [de Swart, 2000] and [Bonami, 2002].

The implementation of aspect in the grammar interacts with many elements that are sensitive to aspect: many verbs, which impose aspectual constraints on their complements (some examples are the progressive auxiliary, which combines with processes, but also verbs like *stop* and *finish*); durational adverbials (*for* adverbials, which combine with processes, and *in* adverbials, which combine with culminated processes, are widely studied with respect to this phenomenon), tenses (as just briefly illustrated), etc.

A full description of the semantics of all tenses implemented in the grammar would be tedious, but an example with the present tense can also be presented. A sentence like *O Samuel tem um filho* "Samuel has a son" receives an MRS representation along the following lines:

> $at(e\{state: +\},\ t) \ \wedge\ includes(t,\ t2\{t\text{-}value:\ utterance\text{-}time\}) \ \wedge$
> $aspectual\text{-}operator(e,\ e2,\ ter(e2,\ X))$, where X is the representation for the verb's arguments.

Here t is the event time, and $t2$ is the utterance time. The present tense is assumed to be an imperfective tense, similar to the past imperfective tense mentioned above: it is associated with an overlap relation, and constrains the clause where it occurs to describe a state. Like Discourse Representation Theory, we assuming that the semantic present is special in that this overlap relation is more specific than just overlap, and it is an inclusion relation: the event time includes the utterance time. Because the verb *ter* "ter" is a state lexically, this is another example where the aspectual operator involved is the identity function.

12.4 Backshift

LXGram contains an implementation of backshift for Portuguese inspired by the work reported in [Costa and Branco, 2012] and [Costa, 2013].

The following pairs of sentences, adapted from [Michaelis, 2006], illustrate the phenomenon of backshift in English, visible in indirect speech. Each sentence in parentheses is the direct speech counterpart of the embedded clause in the same line:

(175) a. Debra said she **liked** wine. ("I **like** wine")
b. Debra said she **likes** wine. ("I **like** wine")
c. Debra said she **brought** the wine. ("I **brought** the wine")
d. Debra said she **had brought** the wine. ("I **brought** the wine")

When the matrix verb is a past tense form, the verb tenses found in the embedded clauses are sometimes different from the tenses used in direct speech (175a, 175d), but not always (175b, 175c). For instance, in this context we sometimes find the simple past instead of the simple present in English (175a). In this respect English is in sharp contrast with Russian, where present tense can be used in similar embedded contexts with the same meanings as the English sentences using the simple past (example from [Schlenker, 2004]):

(176) Petya skazal, čto on plačet. (present tense in the embedded clause)
Petya said that he was crying.

The same backshift phenomenon is also visible in Portuguese:

(177) a. A Debra disse que **gostava** de vinho. ("**Gosto** de vinho")
b. A Debra disse que **gosta** de vinho. ("**Gosto** de vinho")
c. A Debra disse que **trouxe** o vinho. ("**Trouxe** o vinho")
d. A Debra disse que **tinha trazido** o vinho. ("**Trouxe** o vinho")

An initial observation is thus that English and Portuguese use tense in an absolute way (the embedded past tense in (175a) is used to locate a situation in the past), whereas Russian uses it in a relative way (the embedded present tense in (176) marks a situation that was present at the time that the situation in the matrix clause held). Based on similar data, [Comrie, 1986] argues that English exclusively uses tense in an absolute way. However, the example in (178), from [Rodríguez, 2004], shows that in some cases English also uses tense in a relative way. In this example, the past tense is associated with a situation that may hold in the future with respect to the speech time. The past tense here signals precedence with respect to the time of the event in the higher clause (which is in the future). The phenomenon is thus more complicated than a simple separation between languages that use tense in a relative fashion and languages that use it in an absolute manner.

(178) María **will tell** us after the party tomorrow that she **drank** too much.

The same is true of Portuguese:

(179) A Maria **dir**-nos-**á** amanhã depois da festa que **bebeu** demais.

Several verbs trigger tense shifts in their complement. Reporting verbs are often identified with this group, but other verbs, like belief verbs or verbs like *decide* or *remember*, create similar contexts.

The phenomenon is also known as transposition, sequence of tenses or *consecutio temporum*, although some authors use some of these expressions in a broader sense, encompassing constraints on the co-occurrence of tenses in the same sentence. We reserve the term backshift to refer to the more specific case of the complements of the class of verbs just mentioned. We focus on backshift, in this narrow sense. This is because backshift is more constrained than the general co-occurrence of different tenses in the same sentence. For instance, [Rodríguez, 2004] points out that relative clauses are temporally independent, as illustrated by the example in (180a). The same can be observed in Portuguese, as in (180b).

(180) a. Felipe spoke last night with a girl that was crying this morning.
b. O Filipe falou ontem à noite com uma rapariga que estava a chorar hoje de manhã.

Here, two past tenses are found, and the verb of the relative clause refers to a situation that temporally follows the one denoted by the matrix verb. In turn, in backshift contexts involving two past tense forms, the embedded tense never signals a time that temporally follows the time associated with the embedding tense, as the ungrammaticality of the sentence in (181) shows:

(181) * A Debra disse ontem à noite que trouxe uma garrafa de vinho hoje de manhã.
Debra said last night that she brought a bottle of wine this morning.

In LXGram, backshift is treated as the result of the combination of two dimensions. The first one is acknowledging that tense, as it is visible in morphology, is ambiguous, as argued above in Section 12.2. The second dimension consists in classifying the meanings of the tenses along a number of lines: present vs. past vs. future; perfective vs. imperfective aspect, relative vs. absolute. The first two lines determine which kinds of temporal relations are involved in the meaning of tenses (inclusion, overlap or precedence relations), as we have just seen in the previous sections. The third line is how the arguments in these relations are chosen: absolute tenses always take the speech time as one of the arguments of one of these relations; relative times look at a perspective point, which can be the speech time or the time of another event, depending on the syntactic context.

This analysis contains novel aspects. It provides a very clean distinction between absolute and relative tenses, making it depend on the use of two features in its HPSG implementation. It correctly constrains the possible readings of past under past constructions depending on grammatical aspect, which no other theory of backshift explains. This point is mentioned more clearly at the end of the following presentation of our analysis of backshift.

12.4.1 Analysis of Backshift

For the purpose of handling backshift phenomena, we separate semantic tenses into two groups: relative tenses and absolute tenses. The *absolute tenses* always refer to the utterance time directly: they introduce in the semantic representation a temporal relation with the utterance time as one of its arguments. In turn, the *relative tenses* introduce a relation with a perspective point as one of its arguments. This perspective point is the utterance time if the corresponding verb is the head of the main clause of a sentence.[4] This perspective point is instead the event time of a higher verb, if that higher verb is a verb like *say*, triggering backshift.

For the HPSG implementation of such an analysis, revolving around this distinctive constraint of the perspective point and the utterance time, three features are employed: UTTERANCE-TIME, which represents the utterance time, or speech time; PERSPECTIVE-POINT, for this perspective point; and EVENT-TIME, for the event time. As mentioned before we use the type *t* for these features.

The event time is always the second argument of the *at* relation introduced in the MRS representations by the lexical rules responsible for the semantic tenses. These rules add this *at* relation, as well as the remaining relations between temporal indices that we associate with the different tenses, presented above in Table 12.1, in Section 12.2. They also add the relation for the aspectual operators described in Section 12.3.

The utterance time must be accessible at any point in a sentence, because adverbs like *yesterday* or *today* always refer to it (e.g. *today* refers to the day that includes the speech time). In HPSG, each word in a sentence is represented by an instance of the type *word* and each phrase by an instance of the type *phrase*. The feature UTTERANCE-TIME is unified in all words and phrases present in a feature structure representation of a sentence. Therefore, in each phrase, the

[4] This perspective point is similar to the perspective point assumed by DRT. Assuming that, in the case of matrix clauses, the perspective point is always the utterance time is a simplification that we make here because we are only interested in describing backshift (i.e. embedded clauses). The following example, from [Kamp and Reyle, 1993], illustrates the issue:

(1) Mary got to the station at 9:45. Her train would arrive at 10:05.

The perspective point of the second sentence must be the event time of the first sentence, so that this example can be accounted for by saying that conditional verb forms and *would* + infinitive constructions convey a semantic future tense anchored in a past perspective point. More cases where the perspective point of a main clause does not coincide with the utterance time are presented in [Kamp and Reyle, 1993, p.595 and following ones]. Since computational grammars process each sentence in isolation, cases like this are beyond the scope of our work.

UTTERANCE-TIME of the mother node is unified with that of each of its daughters. Similarly, in lexical rules, the UTTERANCE-TIME of the mother node is also unified with the UTTERANCE-TIME of the daughter node. Additionally, in the grammar's start symbol (i.e. the description of what constitutes a valid sentence), the features UTTERANCE-TIME and PERSPECTIVE-POINT are unified: the perspective point is thus the utterance time in matrix clauses.

Because some verbs like *say* trigger backshift in their complement, but other elements do not, the relation between an item's perspective point and that of its complement is controlled lexically, i.e. for each word. For most items (the default case) they are unified, but in the case of backshift triggering elements, the PERSPECTIVE-POINT of the complement is the EVENT-TIME of the head. In HPSG, the lexicon is an association between words and lexical types. The lexical types describe all the grammatical and semantic properties appropriate for the words they are associated with in the lexicon. This relation between a verb's perspective point (or event time) and the perspective point of its complement is encoded in the lexical types.

The absolute tenses look at the feature UTTERANCE-TIME in order to find one of the arguments for the relevant temporal relation that they introduce in the semantics. The relative tenses look at the attribute PERSPECTIVE-POINT instead. As an example, the semantic perfective past tense is a relative tense. Consider the following examples:

(182) a. O Kim mentiu. "Kim lied."
$proper\_q(x,\ named(x,\ "Kim"),$
$at(e_1\{culmination:\ +\},\ t_1)\ \wedge\ before(t_1,\ s)\ \wedge$
$aspectual\text{-}operator(e_1,\ e_2,\ \_mentir\_v(e_2,\ x)))$

b. O Kim disse que mentiu. "Kim said he lied."
$proper\_q(x,\ named(x,\ "Kim"),$
$at(e_1\{culmination:\ +\},\ t_1)\ \wedge\ before(t_1,\ s)\ \wedge$
$aspectual\text{-}operator(e_1,\ e_2,\ \_dizer\_v(e_2,\ x,\ e_3)\ \wedge$
$at(e_3\{culmination:\ +\},\ t_2)\ \wedge\ before(t_2,\ t_1)\ \wedge$
$aspectual\text{-}operator(e_3,\ e_4,\ \_mentir\_v(e_4,\ x))))$

The second argument of the $before$ relation associated with the semantic perfective past is not the utterance time (as has been presented so far) but rather the perspective point, because this tense is a relative tense. In the case of main clauses this perspective point is the utterance time (since the two features utterance-time and perspective-point are unified in the grammar's start symbol)—this is what happens in examples such as (182a), and it is also the case of the matrix verb in (182b). In the case of clauses occurring as the complement of verbs that trigger backshift, this perspective point is the event time of the higher verb. The example in (182b) is thus correctly analyzed as saying that the event of Kim lying precedes the saying event, as can be seen from the semantic representation provided in (182b).

By contrast, the semantic tense given by the English and the Portuguese present tense, in examples like (175b) and (183) below, is an absolute tense.

The semantic present carries an inclusion relation between the event time and another time. Because it is an absolute tense, this other time is always the utterance time, regardless of whether it occurs in backshifted contexts or regular ones. Since the semantic absolute present, being an absolute tense, finds the second argument of the *includes* relation that it adds to the semantic representation in the feature utterance-time, this argument will always be the utterance time, even in backshift contexts, as in (183).[5]

[5] The meaning of the "present under past" is not trivial ([Manning, 1992]), and we opt for a simplified view of it here.

Grammatical Tense	Semantic Tense
Presente (present)	Absolute (imperfective) present
Pretérito imperfeito (imperfective past)	Relative (imperfective) present
Pretérito imperfeito (imperfective past)	Relative imperfective past
Pretérito perfeito (perfective past)	Relative perfective past

Table 12.2: Mapping between some grammatical tenses and some semantic tenses, for Portuguese

(183) O Kim disse que está feliz. “Kim said he is happy.”
$proper\_q(x,\ named(x,\ ''Kim''),$
$at(e_1\{culmination:\ +\},\ t_1)\ \wedge\ before(t_1,\ s)\ \wedge$
$aspectual\text{-}operator(e_1,\ e_2,\ \_dizer\_v(e_1,\ x,\ e_3)\ \wedge$
$at(e_3\{state:\ +\},\ t_2)\ \wedge\ includes(t_2,\ s)\ \wedge$
$aspectual\text{-}operator(e_3,\ e_4,\ \_feliz\_a(e_4, x))))$

We follow the strategy mentioned above of letting a grammatical tense be ambiguous between two or more semantic tenses. The relation between grammatical tense and semantic tense is shown in Table 12.2 for some Portuguese tenses.

The following examples illustrate each of the semantic tenses considered in this table under the influence of a higher past tense verb: the absolute present, denoting overlap with the utterance time, and represented by the Portuguese grammatical present in (184a); the relative present, signaling overlap with the perspective point, and materialized in the Portuguese grammatical imperfective past in (184b); the relative imperfective past, marking precedence with respect to the perspective point, associated with a stative interpretation of the clause and realized by the Portuguese grammatical imperfective past in (184c); and the relative perfective past in (184d), similar to the relative imperfective past but associated with telic situations instead of stative ones and denoted by the Portuguese grammatical perfective past. After each example, under parentheses, one finds its direct speech counterpart.

(184) a. O Kim disse que é feliz. (“Sou feliz”)
Kim said he is happy. (“I am happy”)
Absolute present

b. O Kim disse que era feliz. (“Sou feliz”)
Kim said he was happy. (“I am happy”)
Relative present

c. Ontem o Kim disse que era feliz quando era pequeno. (“Era feliz quando era pequeno”)
Yesterday Kim said he was happy when he was a child. (“I was happy when I was a child”)
Relative imperfective past

d. O Kim disse que já almoçou. (“Já almocei”)
Kim said he already had lunch. (“I already had lunch”)
Relative perfective past

In Portuguese, in backshifted contexts, the grammatical imperfective past is ambiguous: it can be a semantic relative present tense (denoting temporal overlap with the matrix event and corresponding to the grammatical present in direct speech), as in (184b), or a semantic relative imperfective past tense (marking anteriority with respect to the matrix event and corresponding to the grammatical imperfective past in direct speech), as in (184c).

The relative present signals a temporal overlap relation between the time of the event denoted by the verb used in this tense and the perspective point: this is the reading for the example in (184b), where the two events overlap. We give this relative present tense (denoted by grammatical past in backshift contexts) a semantic representation similar to that presented above for the absolute present tense (denoted by grammatical present), the only difference is that the perspective point is used as the second argument of the *includes* relation (it is a relative tense rather than an absolute one). This example is thus analyzed as saying that the event time for the event described in the embedded clause includes the time of the event introduced by the matrix verb.

It must be noted here that our analysis, implemented in LXGram, and just described here, makes a very strong prediction about the relation between perfectivity distinctions and the temporal relation between the two verbs in sentences like the ones in (184). More specifically, an embedded perfective past tense never allows overlap readings with the matrix event, because there is no semantic present tense associated with the grammatical perfective past. This is valid for both Portuguese and English (assuming, like [Michaelis, 2011], that the English grammatical simple past is ambiguous between a semantic perfective past and a semantic imperfective past). No other analysis of backshift found in the literature accounts for this.

Chapter 13

Negative Concord

In Portuguese, most clauses containing multiple negative words are interpreted as containing a single instance of negation. For instance, the following two examples have the same meaning even though they differ in the number of negative words (*não* "not" and *ninguém* "nobody") they contain:

(185) a. Ninguém se feriu no acidente.
nobody himself hurt in the accident
Nobody got hurt in the accident.

b. Não se feriu ninguém no acidente.
not himself hurt nobody in the accident
Nobody got hurt in the accident.

The number of negative words entering negative concord can be higher than two:

(186) Niguém dá nada a ninguém.
nobody gives nothing to nobody
Nobody gives anyone anything (i.e. there are no free lunches).

13.1 Description of the Data and Generalizations

Negative words in Portuguese include:

- *Ninguém* "no one"
- *Nada* "nothing"
- *Não* "not"
- *Nenhum*, *nenhuma*, *nenhuns*, *nenhumas* "no" (determiner), "none"
- *Nunca* "never"

Among these items, sentential negation (*não*) is special in a number of ways (see below).

In each clause, multiple negative words can follow the verb, as in (186) above.

Pre-verbal negative words do not need to occur with sentential negation. They carry negative semantics on their own:

(187) a. Nunca aconteceu.
never it happened
It never happened.

b. Ninguém diz isso.
nobody says that
Nobody says that.

c. Nenhum vi.
no I saw
I saw none.

As the last example shows, grammatical function is not a factor.

In a clause, a negative word can occur after the verb only if another negative word precedes the verb:

(188) a. Não havia nenhum problema.
not there was no problem
There was no problem.

b. * Havia nenhum problema.
there was no problem

Once again, grammatical function is not a factor, only relative position between the negative word and the verb.

One exception to the requirement of negative words before the verb is the preposition phrase *para nada* "for nothing", which can occur after the verb even if no negative word precedes it:

(189) Fomos lá para nada.
we went there for nothing
We went there for nothing.

Most clause types are independent domains for negative concord. That is, it is possible to have multiple negations in one sentence as long as the negations are in different clauses:

(190) a. Quando não há nada, ninguém se preocupa.
when not there is nothing nobody himself cares
When there is nothing, nobody cares.

b. Quem não chora não mama.
who not cries not sucks
Who doesn't cry doesn't get breastfed.

However, negative concord can cross some clause borders. Complement clauses of verbs seem to allow this:

(191) a. Não podiam prever nada.
not they could predict nothing
They couldn't predict anything.

b. Não quero que ninguém se atrase.
not I want that nobody himself gets late
I don't want anybody to be late.

c. Não quero que te falte nada.
not I want that to you be missing nothing
I don't want you to lack anything.

Clauses with zero or only one negative word preceding the verb are fine and easily interpretable. However, speakers vary on their judgments about clauses with multiple negative words before the verb; for some speakers some cases can be viewed as examples of multiple negation:

(192) a. Nunca ninguém o viu.
never nobody him saw
No one has ever seen him.

b. Nunca ninguém pecou.
never nobody sinned
Nobody has ever sinned.

c. Ninguém nunca o viu.
nobody never him saw
No one has (n)ever seen him.

d. Ninguém nunca pecou.
nobody never sinned
Nobody has (n)ever sinned.

Here we will assume that the examples in (192a) and (192b) show negative concord while the examples in (192c) and (192d) are ambiguous and can be interpreted as negative concord or as double negation. We will furthermore assume that this difference is due to the relative word order of the two negative words involved (*ninguém* "nobody" and *nunca* "never"), but it must be pointed out that other factors may be involved in these examples (e.g. pragmatics and common sense knowledge—the double negation reading is easy in an example like (192d)).

Sentential negation (marked with *não*) is special in a number of ways. It cannot iterate, it cannot follow the verb, no negative word can occur between it and the verb (in fact, the only words that can are clitic pronouns), the co-occurrence of sentential negation with other preverbal negative words allows a reading of multiple negation but not a negative concord reading:

(193) a. * Não não aconteceu.
not not it happened

b. * Não aconteceu não.
not it happened not

c. * Aconteceu não.
it happened not

d. * Não ninguém o viu.
not no one him saw

e. Ninguém não o faz.
No one not it does
Nobody does not do it.

Some of these last examples seem to be possible in spoken Brazilian Portuguese, but they are usually not found in written text, so we ignore them.

LXGram contains an implementation of negative concord that accounts for the phenomena just presented. The only aspect that is not covered is the possibility of phrases like *para nada* in post-verbal non-negative contexts, as in (189) above.

13.2 Analysis

The implementation considers negative words (except sentential negation) to be ambiguous between true negative words, which carry negative semantic content and do not need to be licenced by other negative words (we call these *n-words* from now on), and negative polarity items, which carry existential semantics and must be licenced by a negative word (we call them *NPIs* in this discussion). Sentential negation (*não*) is not ambiguous: it is always an n-word, never a negative polarity item.

To this end, the lexicon contains two lexical entries for each negative item, one entry for an n-word and another one for an NPI. The implementation controls the contexts where these elements can occur: broadly speaking, n-words must precede the main verb in the clause where they occur; NPIs in most cases follow a verb, and require a preceding n-word.

Some motivation for this duplication of the lexical entries for negative words is the observation that some slang words (also starting with *n-*) occur only as negative polarity items, never as n-words. This is the case of *nicles* and *népia* (both mean "nothing"), which contrast with *nada* (which also means "nothing"):

(194) a. Não aconteceu nada.
not happened nothing
Nothing happened.

b. Não aconteceu nicles.
not happened nothing
Nothing happened.

c. Não aconteceu népia.
not happened nothing
Nothing happened.

d. Nada aconteceu.
nothing happened
Nothing happened.

e. * Nicles aconteceu.
nothing happened

f. * Népia aconteceu.
nothing happened

Note that this generalization (n-words precede the verb; NPIs follow the verb) needs some qualification. In particular, when negative concord spans multiple clauses, it is possible to have NPIs preceding the embedded verb provided an n-word precedes the higher verb:

(195) Não queremos que ninguém se magoe.
not we want that nobody himself hurts
We don't want anyone to get hurt.

In these cases, it is also possible to have postverbal NPIs in the embedded clause without there being any preverbal n-words in the embedded clause, as long as the higher clause does contain n-words:

(196) Não queremos que se magoe ninguém .
not we want that himself hurts nobody
We don't want anyone to get hurt.

So the correct generalization would not be that NPIs must follow the verb in the clause where they occur, but rather that they would follow some verb in the whole sentence. But even that is too strong, because of the examples in (192).

However, n-words can also occur (pre-verbally) in a clause embedded in a higher negative context. The following sentence is ambiguous:

(197) Não queremos que nada aconteça.
not we want that nothing happens
We don't want anything/nothing to happen.

Since sentential negation cannot be an NPI, the following sentence only has the multiple negation reading:

(198) Não queremos que não aconteça nada.
not we want that not happens nothing
We don't want nothing to happen.

We distinguish the semantic representations associated with the n-words from those associated with the NPIs in the sense field of the predicate name. For instance, the NPI lexical entry for *nunca* "never" contributes to the semantics an elementary predication with the predicate name *_nunca_a_npi_rel.* The lexical entry for the corresponding n-word has in its semantic represention an elementary predication with the predicate name *_nunca_a_nword_rel.* A tacit requirement is that, in the scoped formulas, any relation with *npi* in its sense field should be under the scope of a relation with *nword* in its sense field. The current implementation does not enforce this requirement.

The relevant generalizations mentioned so far are thus:

- N-words can only occur before the verb in their immediate clause, never after the verb.
- Some n-words (like *ninguém* "nobody") can never occur after another n-word in the same clause; others can (such as *nunca* "never"). This accounts for the behavior in (192), where an NPI reading is always available for the second negative word in the sentence, but an n-word reading is not always available for it.
- Negative polarity items must occur after an n-word in the same negative context (which can be larger than the smallest clause where the NPI in question occurs).

13.3 Implementation

The implementation makes use of a feature POLARITY under CAT describing the current polarity context. It is used to control the occurrence of n-words and NPIs. Several subfeatures, which are explained in this section, are appropriate for POLARITY:

$$\begin{bmatrix} \text{POLARITY} & \begin{bmatrix} \text{N-WORDS} & \textit{diff-list} \\ \text{N-WORDS-BEFORE} & \textit{diff-list} \\ \text{EDGES} & \begin{bmatrix} \text{LEFT-EDGE-AFTER-VERB} & \textit{bool} \\ \text{RIGHT-EDGE-AFTER-VERB} & \textit{bool} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The N-WORDS feature encodes the n-words that are dominated by that *sign* (i.e. that word or phrase). The N-WORDS-BEFORE feature contains information about the n-words that precede that *sign*, and the EDGES features describe whether the *sign* occurs after the verb in the clause or before it or dominates it.

The values for these features are obtained from constraints on the lexical types and from other constraints on the types for the syntax rules.

The general idea is that information about the presence of n-words is kept in the N-WORDS feature. This feature takes a difference list as its value. Most lexical items come with an empty N-WORDS list, since they are not n-words. N-words come in the lexicon with a non-empty N-WORDS list, namely a list containing exactly one element.[1]

In general, syntax rules produce nodes with an N-WORDS list that is the concatenation of the N-WORDS lists of the daughters. Exceptions to this are the syntax rules that produce clauses across whose boundaries negative concord is not possible (in the implementation, all clauses except complement clauses of verbs).

Another feature N-WORDS-BEFORE is also a subfeature of the POLARITY feature. It contains information about the n-words that occur before in the same negation domain. The main purpose of the N-WORDS list is to update the N-WORDS-BEFORE of other words and phrases in the sentence. In lexical items, its value is underspecified.

In general, phrase types constrain the N-WORDS-BEFORE list on the basis of the N-WORDS list of previous elements: in binary phrase types, the N-WORDS-BEFORE list of the right daughter has the same value as the N-WORDS list of the left daughter. Also in general, the N-WORDS-BEFORE of the mother is the same as that of its leftmost daughter. The *sign* for the root of a parse tree has an empty N-WORDS-BEFORE, and therefore so does its leftmost daughter. Exceptions to this are phrase types that generate new negation domains.

In the lexical types, the N-WORDS-BEFORE list is underspecified: its value is to be defined by the syntactic context in which a word occurs, in the way just described.

An NPI requires its own N-WORDS-BEFORE list to be non-empty: they must occur after an n-word in the same negative domain. Additionally, some n-words (like *ninguém* "nobody") require its own N-WORDS-BEFORE list to be empty. Other n-words (such as *nunca* "never") do not constrain their N-WORDS-BEFORE list in any way. These constraints cover the second and third generalization in the list above.

The first generalization must be accounted for by keeping track of the verbs that occur inside a clause.

Under POLARITY, we make use of additional edge features, that encode this information. More specifically, we use the boolean features EDGES|LEFT-EDGE-AFTER-VERB and EDGES|RIGHT-EDGE-AFTER-VERB. Lexical items that are not verbs have these two features unified, but their value is underspecified. Verb forms come in the lexicon with the negative value for the feature LEFT-EDGE-AFTER-VERB and the positive value for the feature RIGHT-EDGE-AFTER-VERB.

Similarly to the features N-WORDS and N-WORDS-BEFORE, these EDGE features are also constrained in syntax rules according to adjacent nodes in the structure. Most syntax rules unify the LEFT-EDGE-AFTER-VERB feature of the mother with that of the leftmost daughter, and RIGHT-EDGE-AFTER-VERB of the mother with that of the rightmost daughter. Each of the daughters (except the leftmost one) has its LEFT-EDGE-AFTER-VERB unified with RIGHT-EDGE-AFTER-VERB of its preceding sibling. The syntax rules that do not impose these constraints are the ones that correspond to boundaries between clauses, since this information about whether there is a preceding verb is only relevant inside the smallest clause where that verb occurs.

With this feature geometry in place, n-words are constrained to only occur before the main verb by requiring their LEFT-EDGE-AFTER-VERB feature to be negative. These constraints cover the generalizations stated above.

[1] Since we never read elements from this list, it does not matter which elements we put in it. We put the boolean + there.

13.4 Advanced Topics

ToDoXXX

- *Nem* and *nem. . . nem* “neither. . . not”
- *Sem* and *sem que* “without”

Chapter 14

Miscelaneous

14.1 Noun Phrases as Temporal Modifiers

Some NPs can appear in the syntactic contexts of verbal adjuncts. The following sentence illustrates it, with the NP "o ano passado" (*last year*) modifying the verb:

(199) O comércio atingiu o ano passado 2400 milhões de marcos.
the commerce reached the year passed 2400 millions of marks
Last year commerce reached 2.4 billion marks.

Note that the PP "no ano passado", headed by the preposition "em" (*in*, *on*, *at*), can appear in the syntactic context of the NP "o ano passado".

In LXGram these cases are analyzed as PPs with a missing preposition. A unary syntactic rule, the *missing-temporal-preposition* rule, takes as daughter an NP and produces a node that is similar to a PP headed by the preposition "em".

The kind of NP that can be the daughter of this rule is heavily restricted. In LXGram, its predicate name is constrained to inherit from *time-span_n_rel*. Currently, the predicates that inherit from this type are *_ano_n_rel* ("year"), *_mês_n_rel* (*month*), and *_semana_n_rel* ("week"). This is implemented by constraining the feature PRED under SYNSEM|LOCAL|CONT|KEYS|KEY to be *time-span_n_rel* (this KEY feature is where the relation for the head noun of an NP is contained; it always contains the relation for the head of a constituent). In addition, the quantifier relation of the daughter is also constrained, via the feature SYNSEM|LOCAL|CONT|KEYS|QUANT-REL (see Section 8.5.2 and Section 8.6.4.1 for explanations of how this feature is used in LXGram). It is constrained with a type for quantifier relations that subsumes the quantifier relation of demonstratives and also that of definite articles in modified NPs.

The current analysis undergenerates. It does not allow for a temporal modifier like "segunda-feira" (*on Monday*), since the NP is not introduced by a demonstrative or a definite article. But note that, unless further constraints are implemented, allowing for an NP like that in this context would also for allow an NP like "ano" (*year*), which is impossible.

14.2 Direct Speech

In LXGram, there is a unary syntax rule that takes as daughter a sentence and produces a mother node with constraints similar to complementizer phrases. This syntax rule is called *missing-complementizer*. Furthermore, it requires the daughter sentence to be enclosed in quotes (recall from Section 5.9 that punctuation marks are treated as affixes, and they are recorded in feature structures).

Sentences like the following are analyzable via this construction. The fact that the embedded sentence occurs initially is made possible by the topicalization rule (i.e. this sentence is analyzed by the grammar as having the sentential complement of "dizer" topicalized).

(200) 'O pacto não passa de um sonho', disse à Reuters o comentador turco.
the pact not passes of a dream said to Reuters the commentator Turkish
'The pact is nothing but a dream', said the Turkish commentator to Reuters.

14.3 "Por cento"

The following examples illustrate some syntactic properties of noun phrases containing the expression "por cento".

These NPs are underspecified for number:

(201) a. 43 por cento não votou.
43 percent not voted.SINGULAR
43 percent didn't vote.

b. 43 por cento não votaram.
43 percent not voted.PLURAL
43 percent didn't vote.

The expression "por cento" often, but not necessarily, co-occurs with a PP headed by the preposition "de". This PP provides the domain of quantification:

(202) a. 43 por cento dos portugueses não votou.
43 percent of the Portuguese not voted.SINGULAR
43 percent of the Portuguese didn't vote.

b. 43 por cento dos portugueses não votaram.
43 percent of the Portuguese not voted.PLURAL
43 percent of the Portuguese didn't vote.

The highest noun in this PP does not have to agree with the cardinal numeral preceding "por cento". That numeral is always masculine:

(203) 42 por cento da população não votou.
42$_{\text{MASCULINE PLURAL}}$ percent of the population$_{\text{FEMININE SINGULAR}}$ not voted
42 percent of the population didn't vote.

Even though the numeral is always masculine, the gender of an NP with "por cento" is always the gender of the noun inside the PP:

(204) a. 30 por cento da população será afectada
30 percent of the population$_{\text{FEMININE}}$ will be$_{\text{SINGULAR}}$ afflicted$_{\text{FEMININE}}$
pela gripe.
by the flu
30 percent of the population will be afflicted with the flu.

b. * 30 por cento da população será afectado
30 percent of the population$_{\text{FEMININE}}$ will be$_{\text{SINGULAR}}$ afflicted$_{\text{MASCULINE}}$
pela gripe.
by the flu

c. * 30 por cento da população serão afectados
30 percent of the population$_{\text{FEMININE}}$ will be$_{\text{PLURAL}}$ afflicted$_{\text{MASCULINE}}$
pela gripe.
by the flu

Grammatical number is independent of the number of the noun in the PP:

(205) 5 por cento dos portugueses deverá ser
5 percent of the Portuguese$_{\text{MASCULINE PLURAL}}$ must$_{\text{SINGULAR}}$ be
vacinado.
vaccinated$_{\text{MASCULINE SINGULAR}}$
5 percent of the Portuguese must take the vaccine.

The domain of quantification cannot be given by a noun (a PP with "de" is required), even though other constructions with similar meaning allow it:

(206) a. * 43 por cento portugueses não votaram.
43 percent Portuguese not voted

b. * 43 portugueses por cento não votaram.
43 Portuguese percent not voted

c. 43 portugueses em (cada) cem não votaram.
43 Portuguese in each one hundred not voted
Out of every 100 Portuguese 43 didn't vote.

Nevertheless an NP with the expression "por cento" can contain noun modifiers:

(207) 43 por cento que estavam descontentes não votaram.
43 percent who were displeased not voted
43 percent who were displeased didn't vote.

The expression "por cento" is unanalyzable:

(208) a. 43 em (cada) cem não votaram.
43 in each one hundred not voted
Out of every 100, 43 didn't vote.

b. * 43 por cada cento não votaram.
43 per each one hundred not voted

c. * 43 por duzentos não votaram.
43 per two hundred not voted

In view of these data, the analysis in LXGram for such expressions is as follows. "Por cento" is treated as a single lexical item. The PP headed by "de" is considered a complement of "por cento". The lexical type for this item is, therefore, defined as having an optional complement of this kind.

Since its syntactic distribution is highly restricted, it is given a separate HEAD type, *percent-word*. A dedicated syntax rule is employed to combine a cardinal with "por cento". This rule constrains the HEAD of the right daughter to be of the type *percent-word*. It also constrains the agreement features of the resulting node in the following way: the NUMBER feature is left underspecified, no matter what the NUMBER value of the cardinal daughter is. The GENDER feature of the cardinal daughter is constrained to be *masculine*. The GENDER feature of the mother node comes from the right daughter: it is percolated from the noun in the PP up to the "por cento" item. The mother node is constrained to also be a cardinal.

This syntax rule projects "por cento" as the sister node of a cardinal, and the resulting phrase is a prenominal. This construction is constrained to only co-occur with a null noun, in order to block examples like the ones in (206).

Chapter 15

Syntax Rules

We present the phrase-structure rules in LXGram relevant in the current implementation phase. They are defined in the files `syntax.tdl` and `rules.tdl`.

- ROOT
  Most root nodes of syntax trees are produced by this rule. This is a unary rule whose daughter is constrained to be headed by a verb and have saturated valence lists. This rule is responsible for adding a *qeq* constraint between the global top and the handle of the main verb. In LXGram complementizers also select for a clausal complement and introduce a similar handle constraint in the MRS representation. For this reason, we use a feature ROOT-COMP whose value is a lexical type for complementizers: the relevant attributes are then simply unified (for instance, the constraints on the COMPS element under ROOT-COMP are unified with the SYNSEM of the daughter of this rule).

  See Figure 15.1.

  Ex.: Choveu.
  rained.THIRD-PERSON-SINGULAR
  It rained.

- PRE_ROOT-SENTENCE
  This rule rewrites a start symbol (a sentence) as a pre-root element and a start symbol. The pre-root element can be an interjection, a greeting element ("olá" - *hi*) or a kind of adverbial ("bom" - *well*). This rule can iterate if there are multiple pre-root elements. Currently, the pre-root element is constrained to have a comma attached to it.

  See Figure 15.2.

  Ex.: Olá, estás bem?
  hi you are okay
  Hi, how do you do?

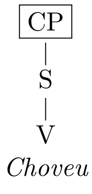


Figure 15.1: The ROOT syntactic rule (corresponding node(s) in a box).

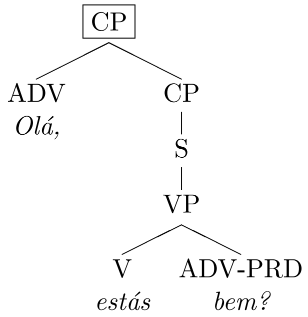


Figure 15.2: The PRE_ROOT-SENTENCE syntactic rule (corresponding node(s) in a box).

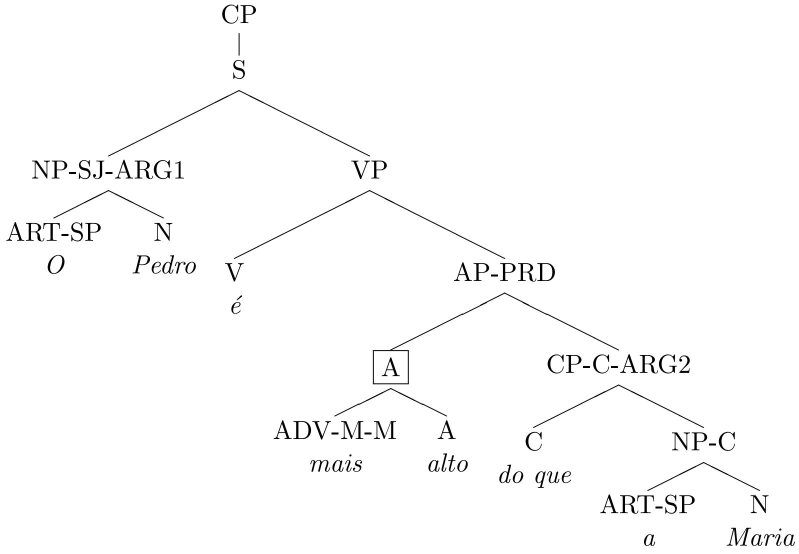


Figure 15.3: The FUNCTOR-HEAD-FCOMPS-ISECT syntactic rule (corresponding node(s) in a box).

- FUNCTOR-HEAD-FCOMPS-ISECT
  This rule projects a functor to the left of its head and unifies the COMPS of the resulting node with those of the functor. See Section 5.3 for details. This rule applies to functors that are given intersective semantics and unifies the LTOP features of both daughters with that of the mother node (see Section 5.3.1). The scopal version of this rule is FUNCTOR-HEAD-FCOMPS-SCOPAL, which only unifies the LTOP of the mother with that of the functor daughter.

  See Figure 15.3.

  Ex.: O Pedro é mais alto do que a Maria.
  the Pedro is more tall than the Maria
  Pedro is taller than Maria.

- FUNCTOR-HEAD-HCOMPS-ISECT
  This rule projects a functor to the left of its head and unifies the COMPS of the resulting node with those of the head. See Section 5.3 for details. This rule applies to functors that are given intersective semantics and unifies the LTOP features of both daughters with that of the mother node (see Section 5.3.1). The scopal version of this rule is FUNCTOR-HEAD-HCOMPS-SCOPAL, which only unifies the LTOP of the mother with that of the functor

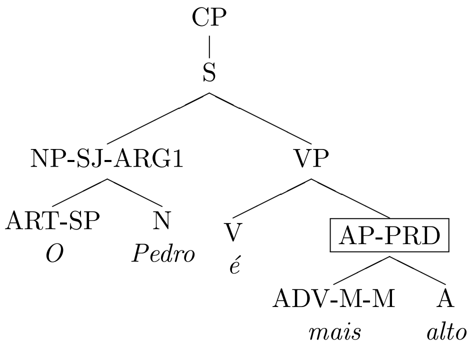


Figure 15.4: The Functor-Head-Hcomps-Isect syntactic rule (corresponding node(s) in a box).

daughter.

See Figure 15.4.

Ex.: O Pedro é mais alto.
the Pedro is more tall
Pedro is taller.

- Head-Functor-Isect
This rule projects a functor to the right of its head and unifies the comps of the resulting node with those of the head. See Section 5.3 for details. This rule applies to functors that are given intersective semantics and unifies the ltop features of both daughters with that of the mother node (see Section 5.3.1). The scopal version of this rule is Head-Functor-Scopal, which only unifies the ltop of the mother with that of the functor daughter.

  See Figure 15.5.

  Ex.: O Pedro era então um consumidor compulsivo de cinema.
  the Pedro was then a consumer compulsive of cinema
  At that time Pedro was a compulsive consumer of movies.

- Subject-Head
This rule is responsible for producing pre-verbal subjects, saturating the verb's subj valence. It will project whichever type of subject a verb selects for. It inherits from *basic-head-subj-phrase* and *head-final*.

  See Figure 15.6.

  Ex.: O carro da Maria é amarelo.
  the car of the Maria is yellow
  Maria's car is yellow.

- Head-Subject
This rule is responsible for rewriting a finite sentence as a subject and a VP. Currently the subject can only be an NP, because verbs selecting for a different kind of subject are not implemented yet, but the rule will rewrite as the the left daughter any kind of element in the head's subj list. It inherits from *basic-head-subj-phrase* in the LinGO Grammar Matrix.

  See Figure 15.7.

  Ex.: Chegou o pai da Maria.
  arrived the father of the Maria
  Maria's father arrived.

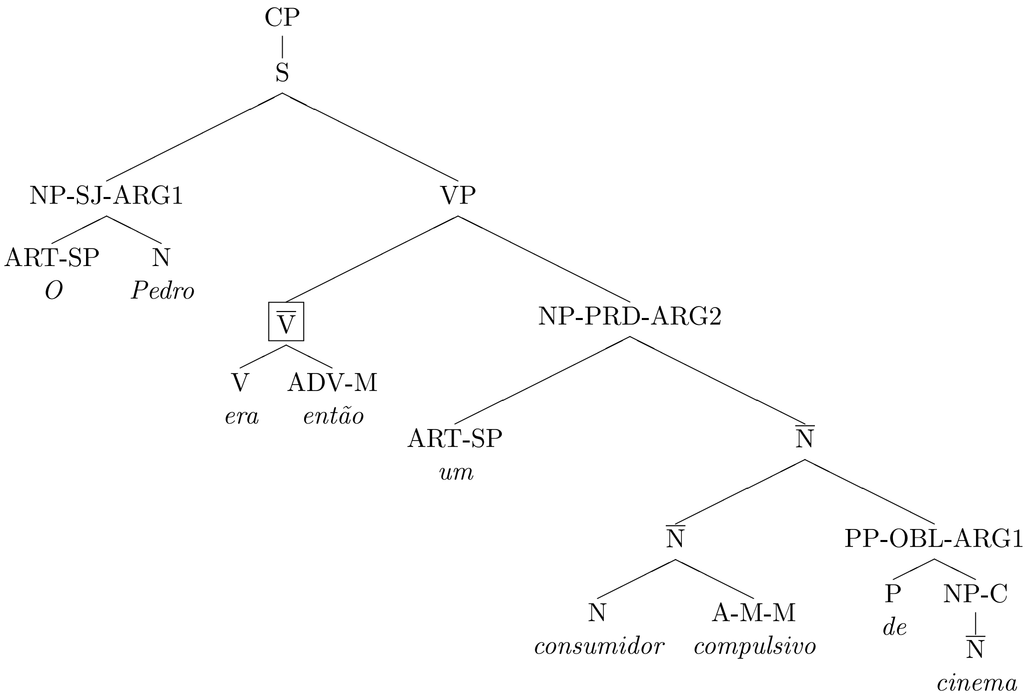


Figure 15.5: The Head-Functor-Isect syntactic rule (corresponding node(s) in a box).

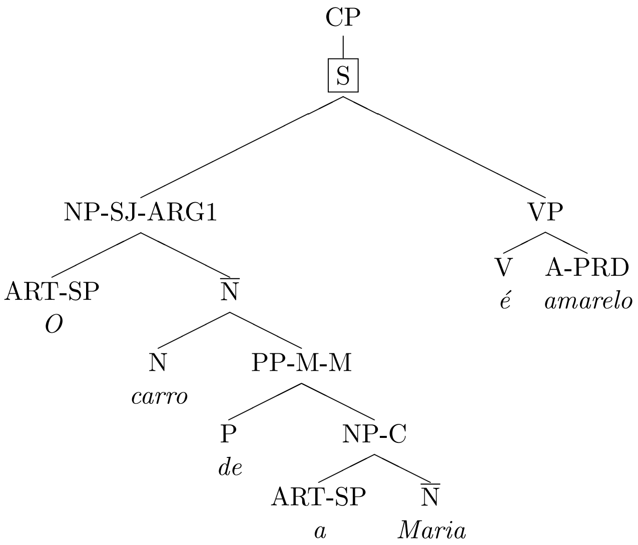


Figure 15.6: The Subject-Head syntactic rule (corresponding node(s) in a box).

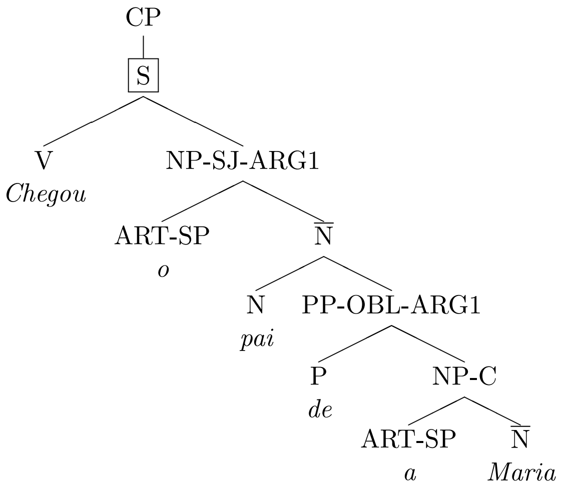


Figure 15.7: The Head-Subject syntactic rule (corresponding node(s) in a box).

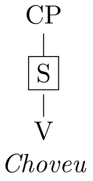


Figure 15.8: The Null_Expletive_Subject syntactic rule (corresponding node(s) in a box).

- Null_Expletive_Subject
  This rule is responsible for discharging an expletive subject from a verb's valence in the cases where the expletive ("ele") is not realized.

  See Figure 15.8.

  Ex.: Choveu.
  rained.THIRD-PERSON-SINGULAR
  It rained.

- Null_Subject-Head
  This rule is responsible for producing non-expletive null subjects. It empties the subj list of a verb and it also adds the semantics of a personal pronoun linked to the argument in the verbal relation that corresponds to the subject. This personal pronoun's agr features are specified only with the information provided by the inflection for subject agreement on the verb.

  See Figure 15.9.

  Ex.: Avariaram.
  broke down.THIRD-PERSON-PLURAL
  They broke down.

- Head-Comp_NotClitic
  This is a binary rule projecting the first complement of any head, and passing the remaining complements up to the mother node. It only applies to non-clitic complements. Currently, there are rules for clitics implemented in LXGram. They are not listed here, because the current implementation of clitics is still experimental.

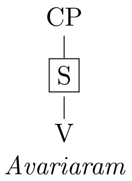


Figure 15.9: The Null_Subject-Head syntactic rule (corresponding node(s) in a box).

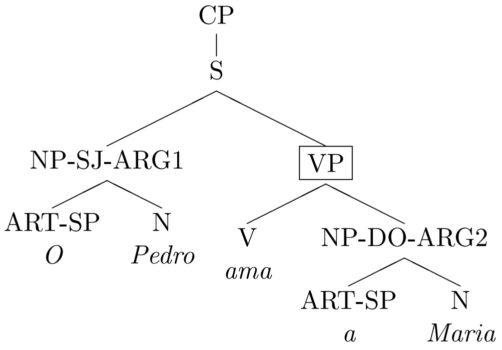


Figure 15.10: The Head-Comp_NotClitic syntactic rule (corresponding node(s) in a box).

See Figure 15.10.

Ex.: O Pedro ama a Maria.
the Pedro loves the Maria
Pedro loves Maria.

- Bare-NP
This rule produces bare NPs. See Section 8.8

See Figure 15.11.

Ex.: Ele bebe vinho.
he drinks wine
He drinks wine.

- Missing-Noun-Head-Initial-Isect
This unary rule produces noun-headed projections from a constituent that can modify a noun. There are four such rules in LXGram: Missing-Noun-Head-Final-Isect, Missing-Noun-Head-Final-Scopal, Missing-Noun-Head-Initial-Isect, Missing-Noun-

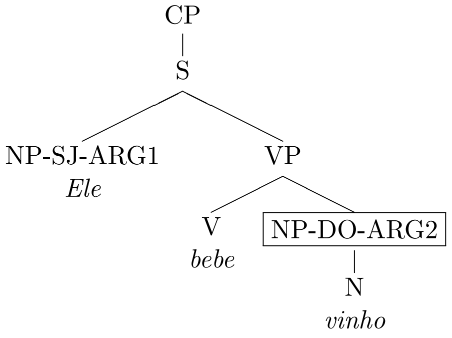


Figure 15.11: The Bare-NP syntactic rule (corresponding node(s) in a box).

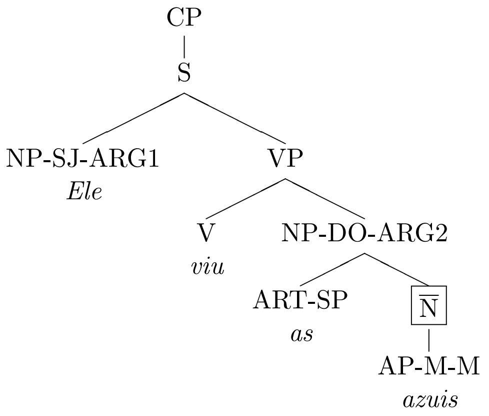


Figure 15.12: The Missing-Noun-Head-Initial-Isect syntactic rule (corresponding node(s) in a box).

Head-Initial-Scopal. The first two rules correspond to the cases when the missing noun would follow the modifier that feeds the rule; the other two are for the cases when the missing noun would precede it. There are two rules for intersective modifiers and two rules for scopal modifiers and specifiers, just like for the Head-Functor rules above. See Section 8.7

See Figure 15.12.

Ex.: Ele viu as azuis.
he saw the blue
He saw the blue ones

- Subject-Head_Extracted
This rule is a subtype of *basic-extracted-subj-phrase* in the LinGO Grammar Matrix. It removes the element of the subj list of its daughter and introduces a slash. See Chapter 11

  See Figure 15.13.

  Ex.: As cartas que chegaram são para ti.
the letters that arrived are for you
The letters that arrived are for you.

- Head-Complement_Extracted
This rule is a subtype of *basic-extracted-comp-phrase* in the LinGO Grammar Matrix. It removes the first element of the comps list of its daughter and introduces a slash. See Chapter 11

  See Figure 15.14.

  Ex.: O carro que compraram é amarelo.
the car that they bought is yellow
The car they bought is yellow.

- Head-Functor-Isect_Extracted
This rule introduces a gap in an adjunct position. The adjunct is an intersective modifier. Details in Chapter 11.

  See Figure 15.15.

  Ex.: Amanhã, estão a dizer que neva.
tomorrow they are saying that it snows
Tomorrow, they're saying it will snow.

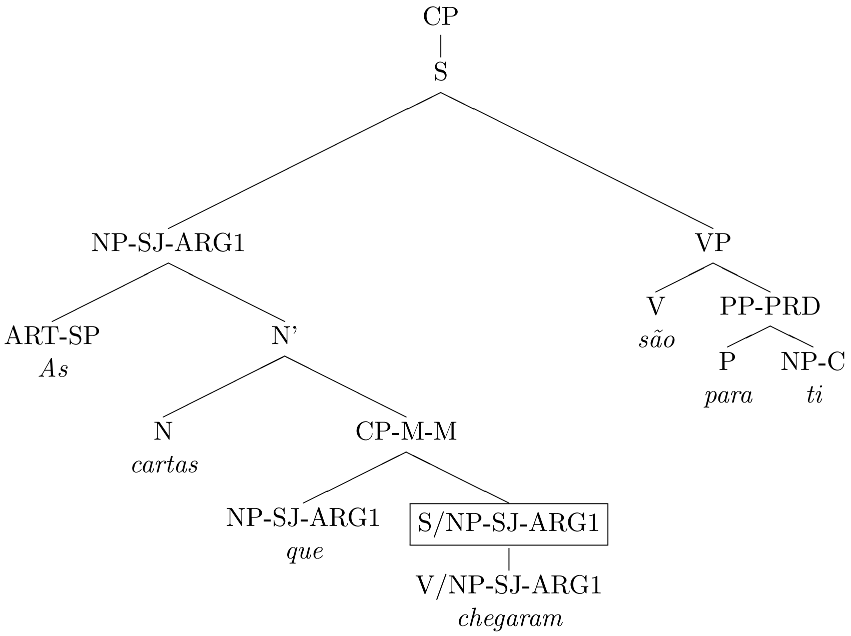


Figure 15.13: The SUBJECT-HEAD_EXTRACTED syntactic rule (corresponding node(s) in a box).

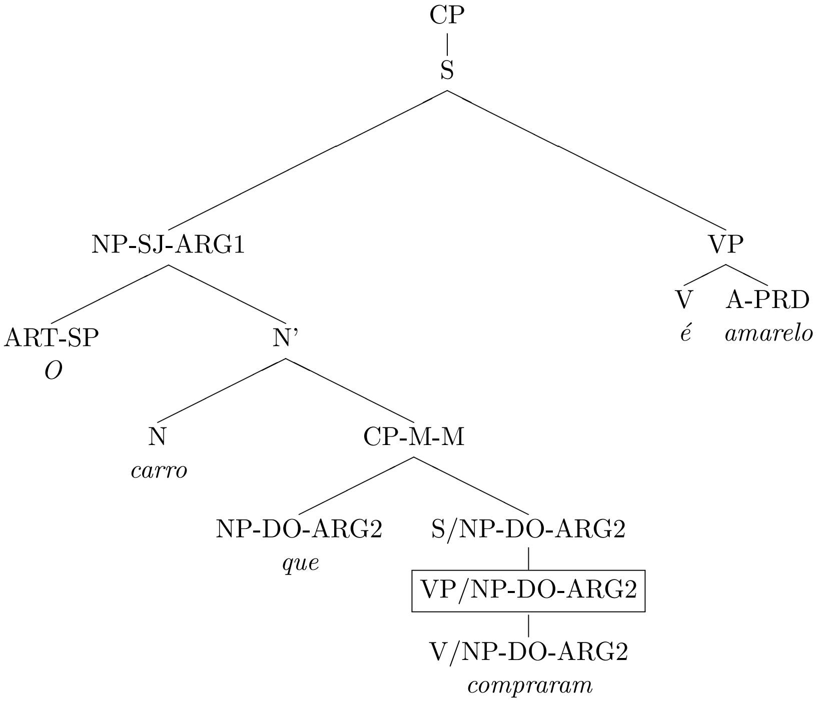


Figure 15.14: The HEAD-COMPLEMENT_EXTRACTED syntactic rule (corresponding node(s) in a box).

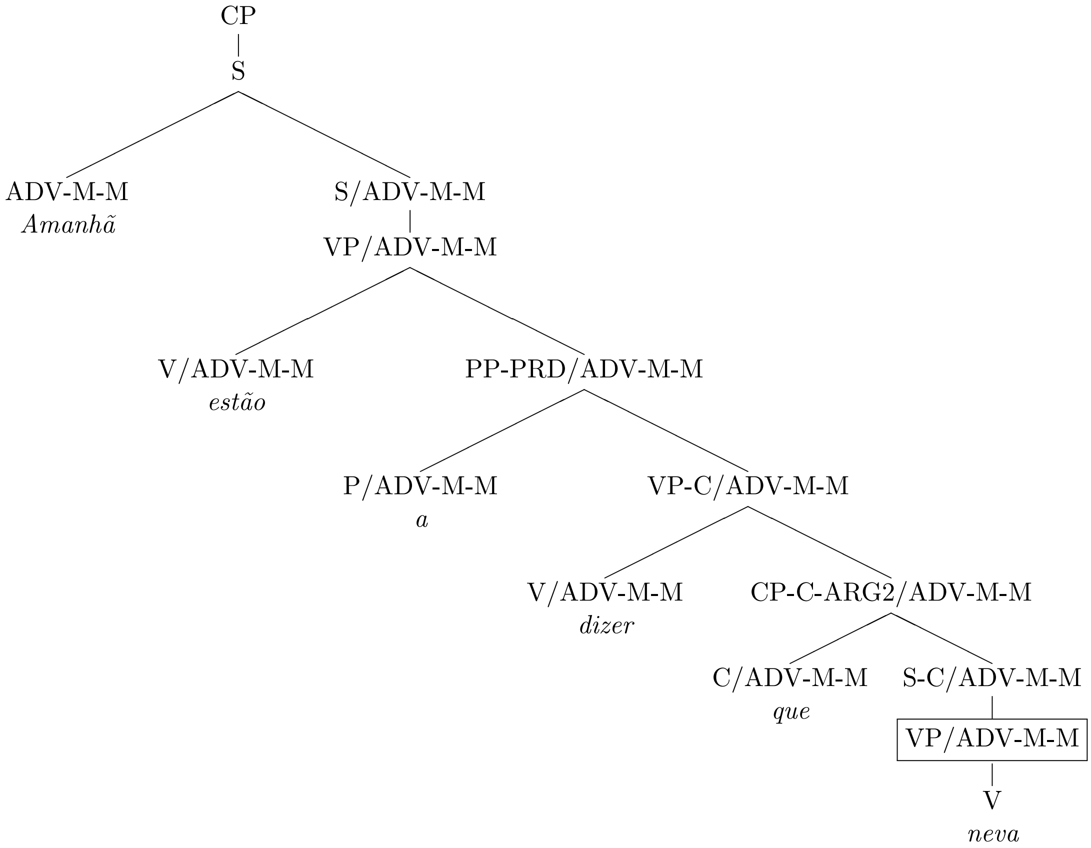


Figure 15.15: The HEAD-FUNCTOR-ISECT_EXTRACTED syntactic rule (corresponding node(s) in a box).

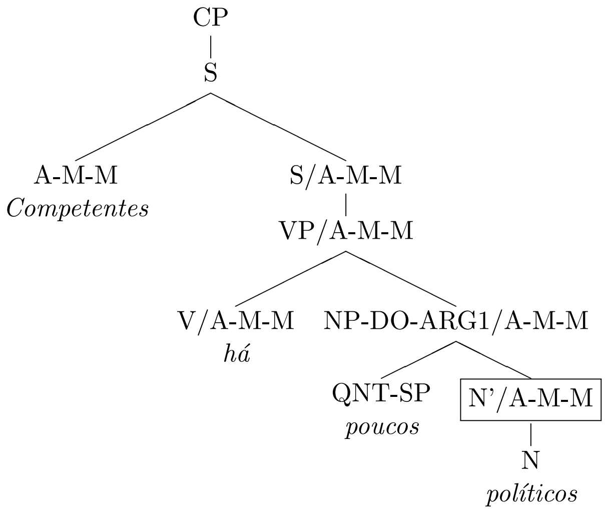


Figure 15.16: The Head-Functor-Scopal_Extracted syntactic rule (corresponding node(s) in a box).

- Head-Functor-Scopal_Extracted
  This rule introduces a gap in an adjunct position. The adjunct is a scopal modifier. Details in Chapter 11.

  See Figure 15.16.

  Ex.: Competentes, há poucos políticos.
  competent they are few politicians
  There are few competent politicians.

- Topic-Head
  This rule introduces a topic on the left of a sentence. Details in Chapter 11.

  See Figure 15.17.

  Ex.: Competentes, há poucos políticos.
  competent they are few politicians
  There are few competent politicians.

- Clause_Relative
  This rule introduces a relative filler at the left periphery of a relative clause. The mother node is a noun modifier. Details in Chapter 11.

  See Figure 15.18.

  Ex.: O carro que compraram é amarelo.
  the car that they bought is yellow
  The car they bought is yellow.

- Free-Relative-Clause
  This rule introduces a relative filler at the left periphery of a relative clause. The mother node is a noun phrase. Details in Chapter 11.

  See Figure 15.19.

  Ex.: Quem não tem cão caça com gato.
  who not has dog hunts with cat
  Who doesn't have a dog goes hunting with a cat.

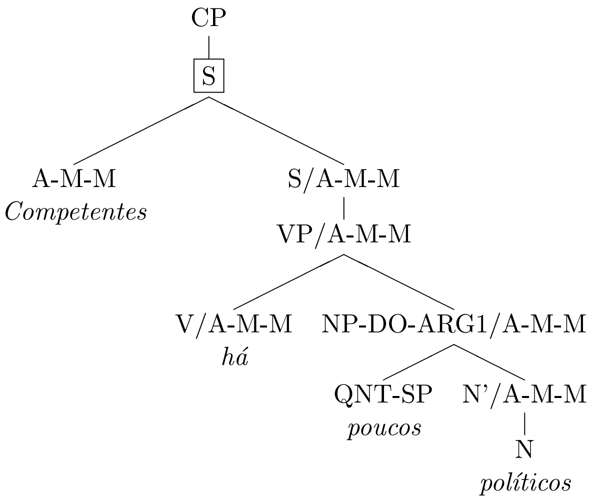


Figure 15.17: The TOPIC-HEAD syntactic rule (corresponding node(s) in a box).

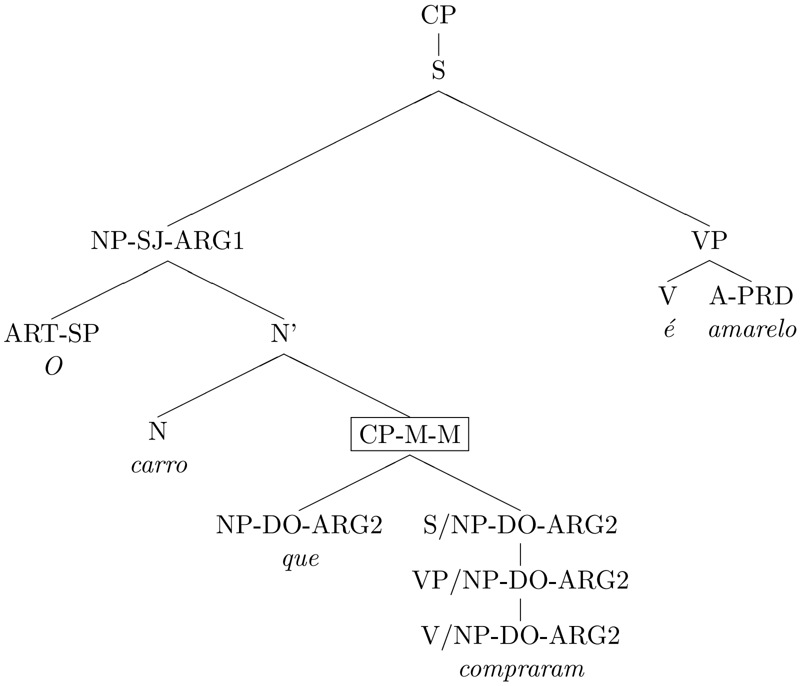


Figure 15.18: The CLAUSE_RELATIVE syntactic rule (corresponding node(s) in a box).

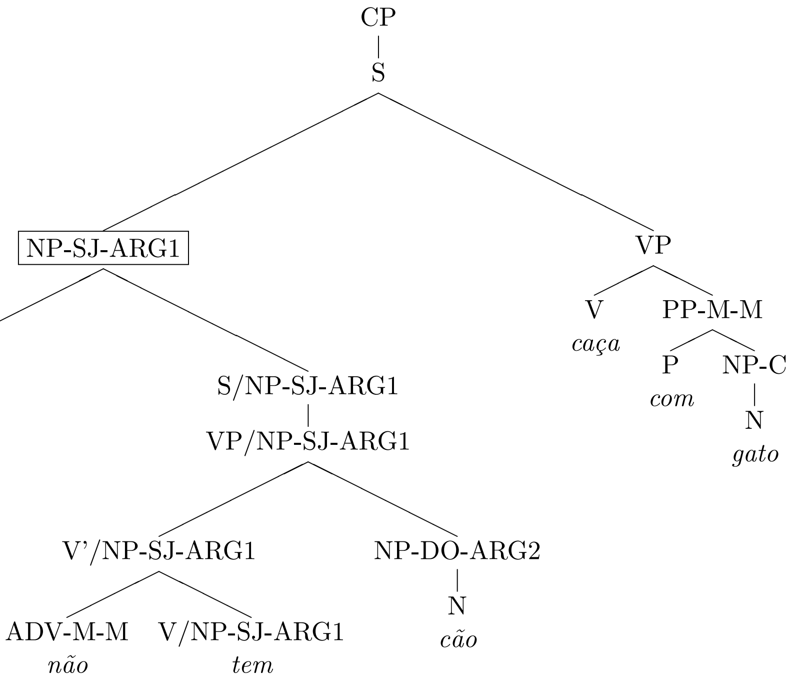


Figure 15.19: The FREE-RELATIVE-CLAUSE syntactic rule (corresponding node(s) in a box).

- NP-COORDINATION-TOP
  This is the top coordination rule for NP coordination. Its left daughter can be an NP or a left coordination rule (for NPs). Its right daughter can be a mid NP coordination rule or a bottom NP coordination rule. Note that the mother node is not an NP but rather an N', in view of the fact that N' modifiers can scope over a coordination of NPs. There are other top coordination rules for: nouns, sentences, verb phrases, verbs, modifiers (also applies to elements in predicative position), nonpredicational PPs. Details in Chapter 10.

  See Figure 15.20.

  Ex.: Não votaram Portugal e Espanha.
  not voted Portugal and Spain
  Portugal and Spain didn't vote.

- NP-COORDINATION-MID
  This is the mid coordination rule for NP coordination. Its left daughter can be an NP or a left coordination rule (for NPs). Its right daughter can be a mid NP coordination rule or a bottom NP coordination rule. There are other mid coordination rules for: nouns, sentences, verb phrases, verbs, modifiers (also applies to elements in predicative position), nonpredicational PPs. Details in Chapter 10.

  See Figure 15.21.

  Ex.: Não votaram Portugal, Espanha e França.
  not voted Portugal Spain and France
  Portugal, Spain and France didn't vote.

- NP-COORDINATION-BOTTOM
  This is the bottom coordination rule for NP coordination. Its left daughter must be a conjunction carrying semantics. Its right daughter must be an NP. There are other bottom

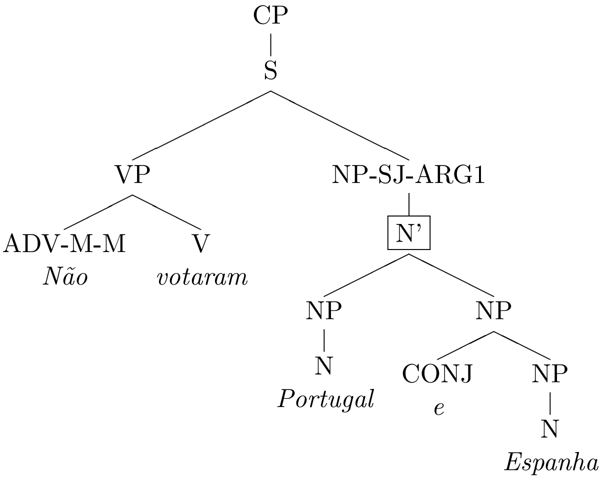


Figure 15.20: The NP-Coordination-Top syntactic rule (corresponding node(s) in a box).

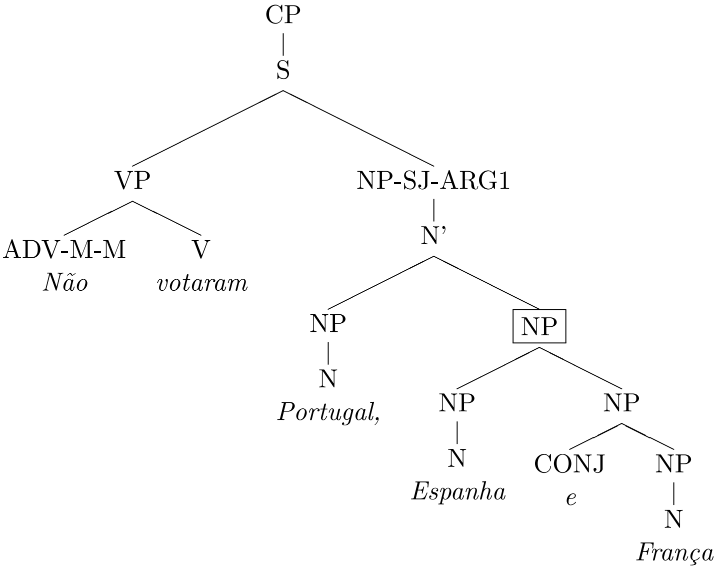


Figure 15.21: The NP-Coordination-Mid syntactic rule (corresponding node(s) in a box).

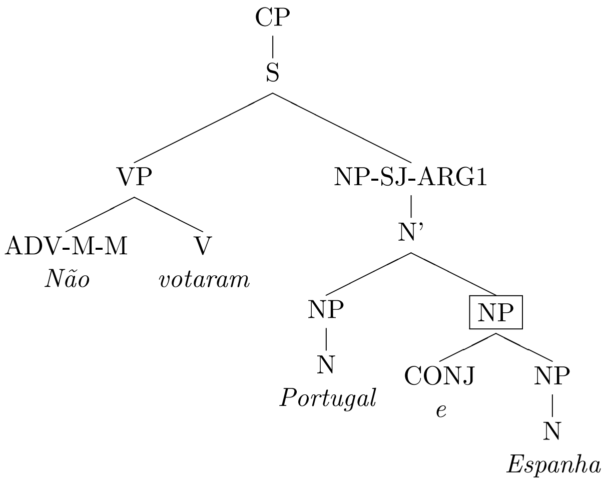


Figure 15.22: The NP-Coordination-Bottom syntactic rule (corresponding node(s) in a box).

coordination rules for: nouns, sentences, verb phrases, verbs, modifiers (also applies to elements in predicative position), nonpredicational PPs. Details in Chapter 10.

See Figure 15.22.

Ex.: Não votaram Portugal e Espanha.
not voted Portugal and Spain
Portugal and Spain didn't vote.

- NP-Coordination-Left
This is the left coordination rule for NP coordination. Its left daughter must be a conjunction not carrying semantics. Its right daughter must be an NP. There are other left coordination rules for: sentences, verb phrases, verbs, modifiers (also applies to elements in predicative position), nonpredicational PPs. Nouns seem to not allow coordination with discontinuous conjunctions ("* um ou trabalhador ou estudante"), and therefore there is no left coordination rule for nouns. Details in Chapter 10.

  See Figure 15.23.

  Ex.: Não votaram ou Portugal ou Espanha.
  not voted either Portugal or Spain
  Either Portugal or Spain didn't vote.

- Missing-Temporal-Preposition
This unary syntax rule takes a noun phrase as its daughter. The mother node has the constraints that are typical of PPs. Details in Section 14.1

  See Figure 15.24.

  Ex.: Este ano o inverno chega mais cedo.
  this year the winter arrives more early
  This year, winter is coming earlier.

- Missing-Complementizer
This unary syntax rule takes a sentence enclosed in quotation marks as its daughter. The mother node has the constraints that are typical of CPs. Details in Section 14.2

  See Figure 15.25.

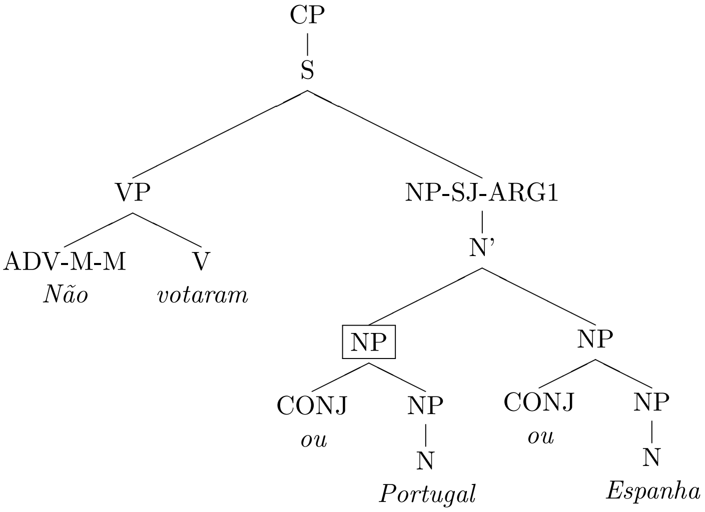


Figure 15.23: The NP-Coordination-Left syntactic rule (corresponding node(s) in a box).

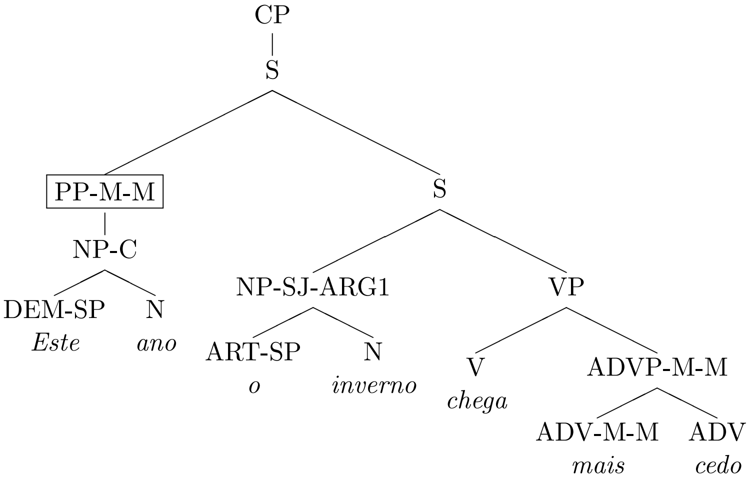


Figure 15.24: The Missing-Temporal-Preposition syntactic rule (corresponding node(s) in a box).

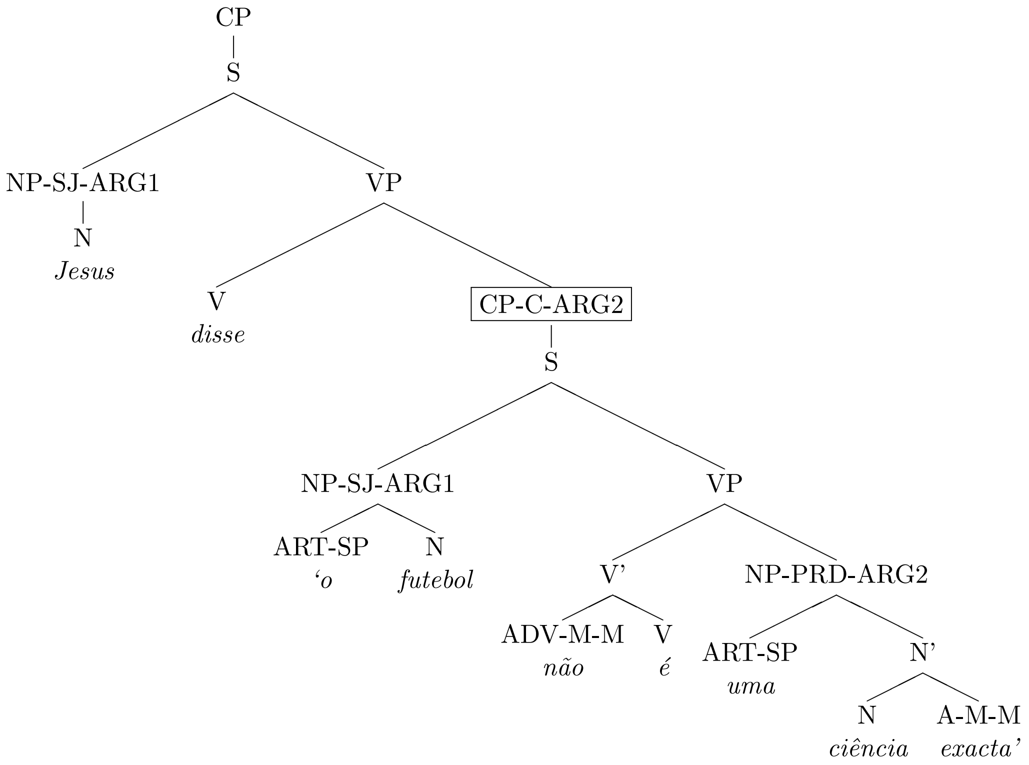


Figure 15.25: The Missing-Complementizer syntactic rule (corresponding node(s) in a box).

Ex.:	Jesus	disse	'o	futebol	não	é	uma	ciência	exacta'.
	Jesus	said	the	football	not	is	a	science	exact

Jesus said 'football isn't an exact science'.

- Name-Name
  This binary rule takes a proper name as its left daughter and either a proper name or another instance of the name-name construction as its right daughter. See Section 8.5.3

  See Figure 15.26.

Ex.:	Chegou	a	Maria	João
	arrived	the	Maria	João

  Maria João arrived.

- NBar-Name
  This binary rule takes a constituent headed by a common noun as its left daughter and an element headed by a proper name as its right daughter. It combines the semantics of the two elements in an intersective way.

  See Figure 15.27.

Ex.:	Chegou	o	senhor	Francisco
	arrived	the	Mr.	Francisco

  Mr. Francisco arrived.

- Null-Precardinal
  This rule takes as daughter a cardinal word and adds *at least* semantics to that cardinal (in the absence of cardinal modifiers like "exactamente"/*exactly*, etc.). The mother has a different head type, so that cardinal modifiers can no longer attach. See Section 8.6.5.1.

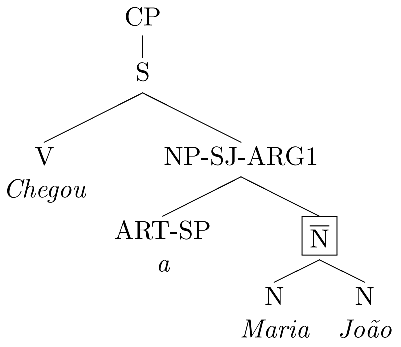


Figure 15.26: The Name-Name syntactic rule (corresponding node(s) in a box).

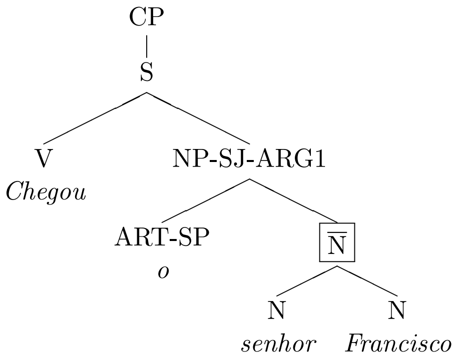


Figure 15.27: The NBar-Name syntactic rule (corresponding node(s) in a box).

See Figure 15.28.

Ex.: Chegaram as três cartas.
arrived the three letters
The three letters arrived.

- Number-Expression-To-Cardinal
This rule takes as daughter a cardinal word (possibly modified by cardinal modifiers) and adds on the mother node constraints that allow it to modify a noun-headed element as a post-determiner. It adds the *cardinal_rel* relation to the semantics. See Section 8.6.5.1.

  See Figure 15.29.

  Ex.: Chegaram as três cartas.
  arrived the three letters
  The three letters arrived.

- Cardinal-to-Determiner
This rule takes as daughter a cardinal post-determiner and, in the absence of a realized determiner, constrains the mother node to attach as a determiner. It adds a quantifier relation to the semantics. See Section 8.6.5.1.

  See Figure 15.30.

  Ex.: Chegaram três cartas.
  arrived three letters
  Three letters arrived.

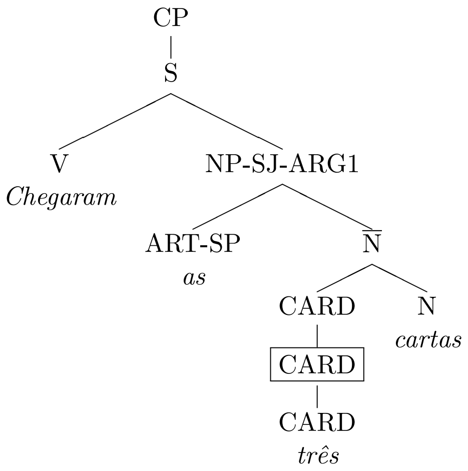


Figure 15.28: The Null-Precardinal syntactic rule (corresponding node(s) in a box).

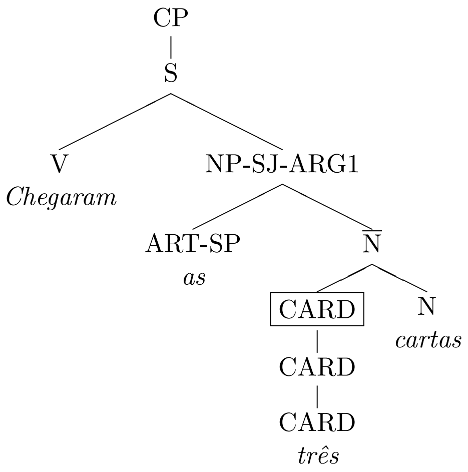


Figure 15.29: The Number-Expression-To-Cardinal syntactic rule (corresponding node(s) in a box).

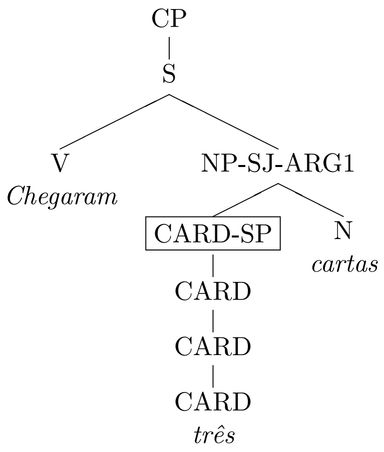


Figure 15.30: The Cardinal-to-Determiner syntactic rule (corresponding node(s) in a box).

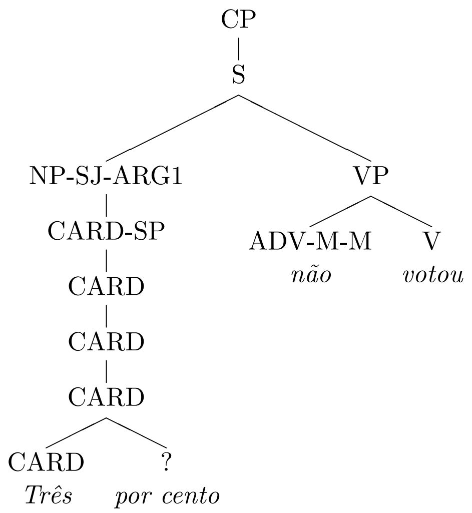


Figure 15.31: The Cardinal-Plus-Percent syntactic rule (corresponding node(s) in a box).

- Cardinal-Plus-Percent
  This rule takes as left daughter a cardinal and as right daughter an expression headed by "por cento" (*percent*). Details in Section 14.3.

  See Figure 15.31.

  Ex.: Três por cento não votou.
  three percent not voted
  Three percent didn't vote.

Chapter 16

Lexical Rules

The lexical rules are defined in the files `irules.tdl` (inflectional rules)and `lrules.tdl` (non-inflectional rules). The file `morphology.tdl` contains the supertypes of the inflectional rules, and `lexical-rules.tdl` is where the supertypes of the non-inflectional lexical rules are defined. The rules for punctuation are defined in the file `punctuation.tdl`. This chapter lists the lexical rules relevant in the current implementation phase.

16.1 Verbal Inflection

In the case of verbal inflection, two sets of rules are applied to recognize fully inflected forms. The first set of rules specifies tense, aspect and mood (TAM) information, the second set is responsible for person and number (PN) agreement with the subject. Verbal forms that do not exhibit subject agreement are produced in a single step, via inflectional rules that produce words (this is controlled via subtypes of *sign*, as all items that can be the daughter of a syntactic rule must be compatible with the type *syntactic-sign*, as presented in Section 8.6.1). Verbal forms that show subject agreement are each produced via a TAM rule that outputs a lexeme (they are of a subtype of *sign* that is incompatible with *syntactic-sign*, namely the type *morphological-sign*) and a PN rule that produces a word.

Some form of underspecification is also employed for ambiguous forms, when it is possible (e.g. third person singular indicative present and second person singular affirmative imperative virtually always exhibit the same surface form, but there is no point in providing a similar, underspecified representation for these, since imperative sentences will have a value for the feature SF different from that of the declarative sentences — see Section 5.4).

LXGram is lacking a rule to produce negative imperative forms (e.g. "corre" – *run* – vs. "não corras" - *don't run*). Only gerunds that are VP or $\overline{\text{N}}$ adjuncts are produced at the moment (see Section 8.6.12).

TAM rules constrain the features where morphological information is encoded. There is a second set of rules, listed below in Section 16.2, which read this information and introduce the appropriate relations in the semantic representations. These morphological features are the following:

$$\begin{bmatrix} \textit{sign} & \\ \textsc{morph}|\textsc{verbal-m} & \begin{bmatrix} \textsc{m-tense} & \textit{morph-tense} \\ \textsc{m-mood} & \textit{mood} \end{bmatrix} \end{bmatrix}$$

The feature M-TENSE encodes verb tense, and the feature M-MOOD encodes verb mood.

The feature M-MOOD is structure shared with the feature SYNSEM|LOCAL|CAT|HEAD|VFORM|VMOOD. Mood must be encoded in a feature under SYNSEM as well, because mood can be selected for. For

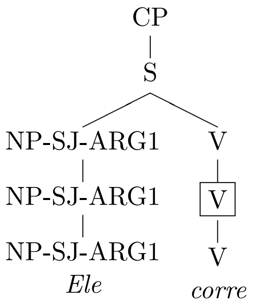


Figure 16.1: The Pres-Ind-Verb lexical rule (corresponding node(s) in a box).

instance, a verb like *insistir* "insist" selects for a CP complement with the verb in the subjunctive mood. The feature vform is present in verbs and also in complementizers, and complementizers unify their vmood feature with that of the head of their complement.

TAM rules constrain the m-tense feature to be a supertype of two types: one for a simple tense, and another type for the compound tense (with the auxiliary verbs *ter* or *haver* "have") which uses the auxiliary in that tense. The auxiliaries have that feature constrained to be of the type *compound-tense*, whereas the other verbs have it constrained to be of the type *simple-tense*. For instance, the rule for present indicative constrains m-tense to be of type *based-on-presente*. The unifier of *based-on-presente* and *simple-tense* is *presente*: when this rule is applied to any verb except the compound tense auxiliaries, it produces a form with the value *presente* for the feature m-tense. The unifier of *based-on-presente* and *compound-tense* is *pretérito-perfeito-composto* (similar to English present perfect): when this rule is applied to an auxiliary, it produces a form with this value for the feature m-tense. Auxiliaries obligatorily select for a past participle, but the participle itself does not impose constraints on this feature, since it already of the desired type on the auxiliary.

The TAM rules that produce lexemes that are to be further inflected for subject agreement are listed next.

- Pres-Ind-Verb

  This rule adds the constraint

  $$\left[\textsc{morph}|\textsc{verbal-m} \left[\begin{array}{ll} \textsc{m-tense} & \textit{based-on-presente} \\ \textsc{m-mood} & \textit{indicativo} \end{array}\right]\right]$$

  See Figure 16.1.

  Ex.: Ele corre.
  he runs
  He runs.

- Pret-Imp-Ind-Verb

  This rule adds the constraint

  $$\left[\textsc{morph}|\textsc{verbal-m} \left[\begin{array}{ll} \textsc{m-tense} & \textit{based-on-pretérito-imperfeito} \\ \textsc{m-mood} & \textit{indicativo} \end{array}\right]\right]$$

  Ex.: Ele corria.
  he ran
  He ran/would run/used to run.

- Pret-Perf-Ind-Verb

This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{pretérito-perfeito}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Ele correu.
he ran
He ran/has run.

- Pres-Or-Pret-Perf-Ind-Verb
This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{based-on-presente-or-pretérito-perfeito}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Nós corremos.
we run/ran
We run/ran.

- Plu-Ind-Verb
This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{based-on-pretérito-mais-que-perfeito}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Ele correra.
he had run
He had run.

- Pret-Perf-or-Pluperf-Ind-Verb
This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{based-on-pretérito-perfeito-or-pretérito-mais-que-perfeito}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Eles correram.
they ran
They ran/have run/had run.

- Fut-Ind-Verb
This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{based-on-futuro}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Ele correrá.
he will run
He will run.

- Fut-Pret-Ind-Verb
This rule adds the constraint

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\begin{array}{ll}\textsc{m-tense} & \textit{based-on-condicional}\\ \textsc{m-mood} & \textit{indicativo}\end{array}\right]\right]$$

Ex.: Ele correria.
he would run
He would run.

- Pres-Subj-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-TENSE} & \textit{based-on-presente}\\ \text{M-MOOD} & \textit{conjuntivo-not\_futuro}\end{bmatrix}\right]$$
  Ex.: Não tenho nenhum que funcione.
  not I have any that works.PRESENT-SUBJUNCTIVE
  I don't have any that works.

- Pret-Imp-Subj-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-TENSE} & \textit{based-on-pretérito-imperfeito}\\ \text{M-MOOD} & \textit{conjuntivo-not\_futuro}\end{bmatrix}\right]$$
  Ex.: Não tinha nenhum que funcionasse.
  not I had any that worked.IMPERFECT-SUBJUNCTIVE
  I didn't have any that worked.

- Fut-Subj-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-TENSE} & \textit{based-on-futuro}\\ \text{M-MOOD} & \textit{conjuntivo-futuro}\end{bmatrix}\right]$$
  Ex.: Comprarei o primeiro que chegar.
  I will buy the first that arrives.FUTURE-SUBJUNCTIVE
  I'll buy the first one to arrive.

- Pres-Imp-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-TENSE} & \textit{based-on-presente}\\ \text{M-MOOD} & \textit{imperativo}\end{bmatrix}\right]$$
  Ex.: Corre.
  run
  (You) run.

- Infl-Inf-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-MOOD} & \textit{infinitivo-flexionado}\end{bmatrix}\right]$$
  Ex.: É para veres.
  it is for you to see
  It is for you to see.

The TAM rules that produce words are these:

- Inf-Verb
  This rule adds the constraint
  $$\left[\text{MORPH}|\text{VERBAL-M}\ \begin{bmatrix}\text{M-MOOD} & \textit{infinitivo-não-flexionado}\end{bmatrix}\right]$$
  See Figure 16.2.

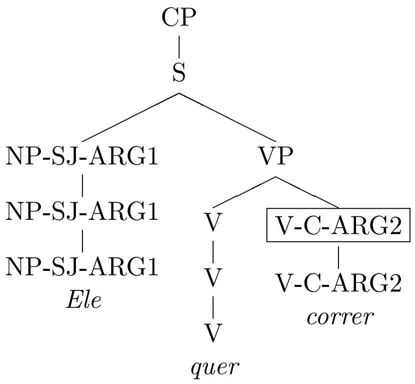


Figure 16.2: The Inf-Verb lexical rule (corresponding node(s) in a box).

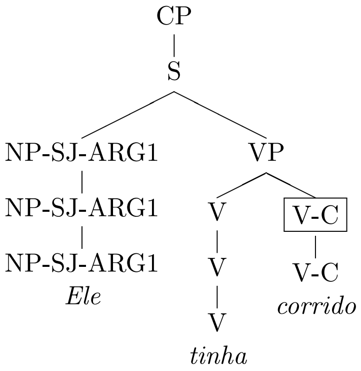


Figure 16.3: The Part-Pass-Inv lexical rule (corresponding node(s) in a box).

Ex.: Ele quer correr.
he want to run
He wants to run.

- Part-Pass-Inv
  This rule adds the constraint
  $\left[\textsc{morph}|\textsc{verbal-m}\ \left[\textsc{m-mood}\quad \textit{particípio-passado-inv}\right]\right]$

  See Figure 16.3.

  Ex.: Ele tinha corrido.
  he had run
  He had run.

- Passive-Participle
  This rule produces the forms of the past participle used in passives. The output of this rule will feed the rules for nominal inflection in gender and number. More details in Section 9.2.1.

  See Figure 16.4.

  Ex.: A cidade foi destruída.
  the city was destroyed
  The city was destroyed.

- Noun-Modifying-Gerund
  This rule adds the constraint

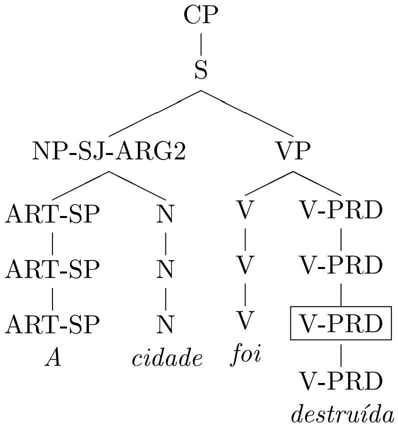


Figure 16.4: The Passive-Participle lexical rule (corresponding node(s) in a box).

$$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\textsc{m-mood}\quad \textit{gerúndio}\right]\right]$$

See Figure 16.5.

Ex.: Ele comprou um livro descrevendo a Irlanda.
he bought a book describing the Ireland
He bought a book describing Ireland.

- Verb-Modifying-Gerund
  This rule adds the constraint
  $$\left[\textsc{morph}|\textsc{verbal-m}\ \left[\textsc{m-mood}\quad \textit{gerúndio}\right]\right]$$
  See Figure 16.6.

  Ex.: Ele saiu correndo.
  he left running
  He left running.

The PN rules are the following:

- 1Sg-OR-3Sg-Verb
  This rule adds the constraint
  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\ \left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\ \begin{bmatrix}\textsc{person} & \textit{first-or-third}\\ \textsc{number} & \textit{singular}\end{bmatrix}\right]\right\rangle\right]$$
  See Figure 16.7.

  Ex.: Corria.
  ran.FIRST-OR-THIRD-PERSON-SINGULAR
  I/he/she/it ran.

- 1Sg-Verb
  This rule adds the constraint
  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\ \left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\ \begin{bmatrix}\textsc{person} & \textit{first}\\ \textsc{number} & \textit{singular}\end{bmatrix}\right]\right\rangle\right]$$

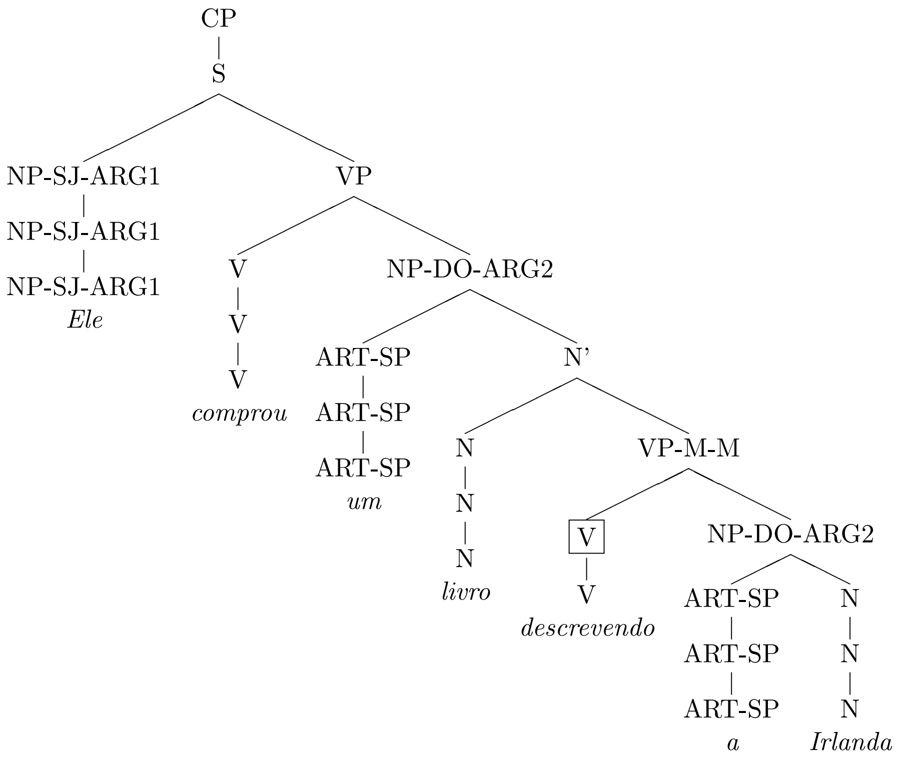


Figure 16.5: The Noun-Modifying-Gerund lexical rule (corresponding node(s) in a box).

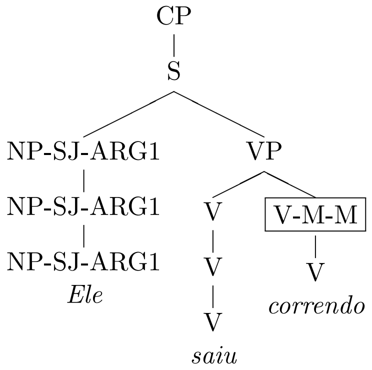


Figure 16.6: The Verb-Modifying-Gerund lexical rule (corresponding node(s) in a box).

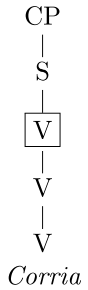


Figure 16.7: The 1Sg-OR-3Sg-Verb lexical rule (corresponding node(s) in a box).

Ex.: Eu corro.

I run

I run.

- 2Sg-Verb

  This rule adds the constraint

  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\begin{bmatrix}\textsc{person} & \textit{second}\\ \textsc{number} & \textit{singular}\end{bmatrix}\right]\right\rangle\right]$$

  Ex.: Tu corres.

  you run

  You run.

- 3Sg-Verb

  This rule adds the constraint

  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\begin{bmatrix}\textsc{person} & \textit{third}\\ \textsc{number} & \textit{singular}\end{bmatrix}\right]\right\rangle\right]$$

  Ex.: Ele corre.

  he runs

  He runs.

- 1Pl-Verb

  This rule adds the constraint

  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\begin{bmatrix}\textsc{person} & \textit{first}\\ \textsc{number} & \textit{plural}\end{bmatrix}\right]\right\rangle\right]$$

  Ex.: Nós corremos.

  we run

  We run.

- 2Pl-Verb

  This rule adds the constraint

  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\begin{bmatrix}\textsc{person} & \textit{second}\\ \textsc{number} & \textit{plural}\end{bmatrix}\right]\right\rangle\right]$$

  Ex.: Vós correis.

  you run

  You run.

- 3Pl-Verb

  This rule adds the constraint

  $$\left[\textsc{synsem}|\textsc{local}|\textsc{cat}|\textsc{val}|\textsc{subj}\left\langle\left[\textsc{local}|\textsc{cat}|\textsc{head}|\textsc{agr}\begin{bmatrix}\textsc{person} & \textit{third}\\ \textsc{number} & \textit{plural}\end{bmatrix}\right]\right\rangle\right]$$

  Ex.: Eles correm.

  they run

  They run.

16.2 Verbal Inflection—The Meanings of the Tenses

LXGram uses an additional set of lexical rules that apply to the output of the lexical rules for verbal inflection. These additional rules add relations to the MRS representation being built that depend on the morphological tense of the verb.

- Present

  This rule takes as input a form whose m-tense feature has the value *presente*.

  It adds present tense semantics to the MRS representation, namely that the event time includes the speech time (or utterance time):

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{includes\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{now6} \begin{bmatrix} \textit{now} & \\ \text{T-VALUE} & \textit{utterance\_time} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Past-Imperfective

  This rule takes as input a form whose m-tense feature has the value *pretérito-imperfeito* (the unifier of *based-on-pretérito-imperfeito* and *simple-tense*) and an m-mood feature with the value *indicativo*.

  It adds imperfective past tense semantics to the MRS representation. Namely, it says that the event time overlaps a past time (i.e. a time that precedes the speech time):

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{overlaps\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t6} \\ \text{ARG1} & \boxed{t7}\ t \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e8}\ e \\ \text{ARG2} & \boxed{h9}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Past-Perfective

  This rule takes as input a form whose m-tense feature has the value *pretérito-perfeito*.

  It adds perfective past tense semantics to the MRS representation. Namely, it says that the event time precedes the speech time:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Pluperfect

  This rule takes as input a form whose m-tense feature has the value *pretérito-mais-que-perfeito.*

  It adds anterior-to-past semantics to the MRS representation. Namely, it says that the event time precedes some past time:

$$\left[\begin{array}{ll} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \left[\begin{array}{ll} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{array}\right], \left[\begin{array}{ll} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{array}\right], \left[\begin{array}{ll} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{array}\right] \right\rangle \end{array}\right]$$

- Future

  This rule takes as input a form whose m-tense feature has the value *presente-or-futuro-simples* (i.e. present of future forms).

  It adds future semantics to the MRS representation. Namely, it says that the event time is after the speech time:

$$\left[\begin{array}{ll} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \left[\begin{array}{ll} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{array}\right], \left[\begin{array}{ll} \textit{is-after\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{now6}\left[\begin{array}{ll} \textit{now} & \\ \text{T-VALUE} & \textit{utterance\_time} \end{array}\right] \end{array}\right], \left[\begin{array}{ll} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{array}\right] \right\rangle \end{array}\right]$$

- Future-of-past-rule

  This rule takes as input a form whose m-tense feature has the value *condicional-simples* (i.e. a conditional form).

  It adds future of past semantics to the MRS representation. Namely, it says that the event time is after a time that precedes the speech time:

$$\left[\begin{array}{ll} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \left[\begin{array}{ll} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{array}\right], \left[\begin{array}{ll} \textit{is-after\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{array}\right], \left[\begin{array}{ll} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t6} \\ \text{ARG1} & \boxed{now7}\ \textit{now} \end{array}\right], \left[\begin{array}{ll} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e8}\ e \\ \text{ARG2} & \boxed{h9}\ h \end{array}\right] \right\rangle \end{array}\right]$$

- Past-of-future-rule

  This rule takes as input a form whose m-tense feature has the value *futuro-composto* (i.e. the Portuguese equivalent to future perfect).

It adds past of future semantics to the MRS representation. Namely, it says that the event time is before a time that is after the speech time:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-after\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t6} \\ \text{ARG1} & \boxed{now7}\ now \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e8}\ e \\ \text{ARG2} & \boxed{h9}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Untensed-finite
  This rule takes as input a form whose m-mood feature has the value *imperativo* (i.e. it is an imperative form).

  It adds no temporal semantics to the MRS representation.

- Untensed-gerund
  This rule takes as input a form whose m-mood feature has the value *gerúndio* (i.e. it is a gerund form).

  It adds no temporal semantics to the MRS representation.

- Untensed-infinitive
  This rule takes as input a form whose m-mood feature has the value *infinitivo* (i.e. it is an infinitive form, either an inflected infinitive or a non-inflected infinitive).

  It adds no temporal semantics to the MRS representation.

- Backshifted-present
  This rule takes as input a form whose m-tense feature has the value *pretérito-imperfeito* and which inside the complement of a backshift-triggering verb.

  It produces a semantic representation that describes the event time as including the event time of the higher verb:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{includes\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Backshifted-future
  This rule applies to forms in the conditional or in the *pretérito-imperfeito* tense (indicative or subjunctive) and occurring inside the complement of a backshift-triggering verb.

  It produces a semantic representation that describes the event time as being after the event time of the higher verb:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-after\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e7}\ e \\ \text{ARG2} & \boxed{h8}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

- Backshifted-past-of-future

  This rule applies to forms in the *condicional composto* and occurring inside the complement of a backshift-triggering verb.

  It produces a semantic representation that describes the event time as being before a time that is after the event time of the higher verb:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{e4}\ e \\ \text{ARG1} & \boxed{t5}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t5} \\ \text{ARG1} & \boxed{t6}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-after\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{t6} \\ \text{ARG1} & \boxed{t7}\ t \end{bmatrix}, \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h3} \\ \text{ARG0} & \boxed{e4} \\ \text{ARG1} & \boxed{e8}\ e \\ \text{ARG2} & \boxed{h9}\ h \end{bmatrix} \right\rangle \end{bmatrix}$$

16.3 Nominal Inflection

Four rules are used to produce all gender and number inflected forms of nouns, pronouns and all noun modifiers/specifiers that inflect for these two categories (determiners, adjectives, possessives, etc.). They are also organized in two sets. The rules in the first one are responsible to constrain gender information and their output is a lexeme, the rules in the second set inflect these elements in number and produce words.

Not all of these elements undergo the gender rules, however, viz. they only do if they show gender distinctions. This information must of course be lexically specified (e.g. "gato" – *cat* – has a feminine version "gata"; "carro" – *car* does not). This is the current implementation of nominal morphology in LXGram. This implementation will change in future versions, in order to better accommodate the integration of an external morphological analyzer.

For adjectives there is another rule to produce "-íssimo" suffixed forms (a kind of superlative), that applies before any of the other rules.

The inflectional rules for nominals are the following:

- Masc-Nominal

  This rule produces the masculine forms. For nouns and adjectives, it applies only to those items that show gender distinctions (which must be stated in the lexicon). In the example below it applies neither to the noun "sofá" (*couch*), which, referring to an inanimate entity, has a single, lexically specified gender, nor to the adjective "verde" (*green*), which does not overtly show gender agreement.

  See Figure 16.8.

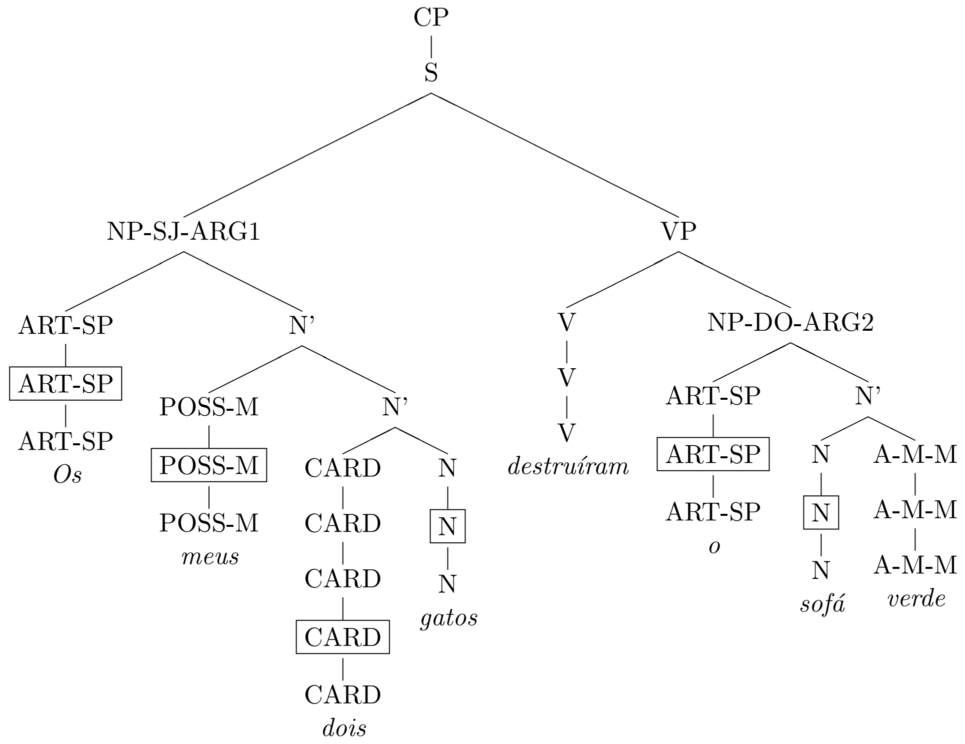


Figure 16.8: The Masc-Nominal lexical rule (corresponding node(s) in a box).

Ex.:	Os	meus	dois	gatos	destruíram	o	sofá	verde.
	the	my	two	cats.MASCULINE	destroyed	the	couch.MASCULINE	green

My two cats destroyed the green couch.

- Fem-Nominal

  This rule produces the feminine forms. For nouns and adjectives, it applies only to those items that show gender distinctions (which must be stated in the lexicon). In the example below it applies neither to the noun "mesa" (*table*), which, referring to an inanimate entity, has a single, lexically specified gender, nor to the adjective "verde" (*green*), which does not overtly show gender agreement.

  See Figure 16.9.

Ex.:	As	minhas	duas	gatas	destruíram	a	mesa	verde.
	the	my	two	cats.FEMININE	destroyed	the	table.FEMININE	green

  My two cats destroyed the green table.

- Sg-Nominal

  This rule produces the singular forms from a gender specified lexeme. It will apply after inflectional rules for gender (most of the nominals in the example below) or directly to a lexical item if it does not inflect for gender ("sofá" and "verde" in the example below).

  See Figure 16.10.

Ex.:	O	meu	gato	destruiu	o	sofá	verde.
	the	my	cat	destroyed	the	couch	green

  My cat destroyed the green couch.

- Pl-Nominal

  This rule produces the plural form from a gender specified lexeme. Since number inflection

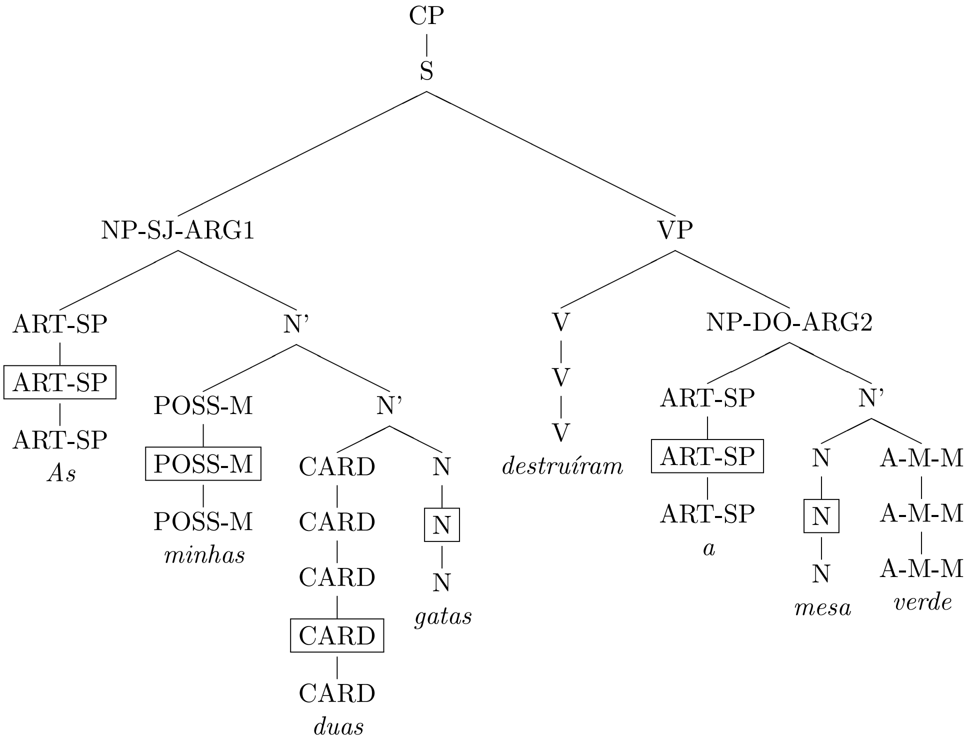


Figure 16.9: The Fem-Nominal lexical rule (corresponding node(s) in a box).

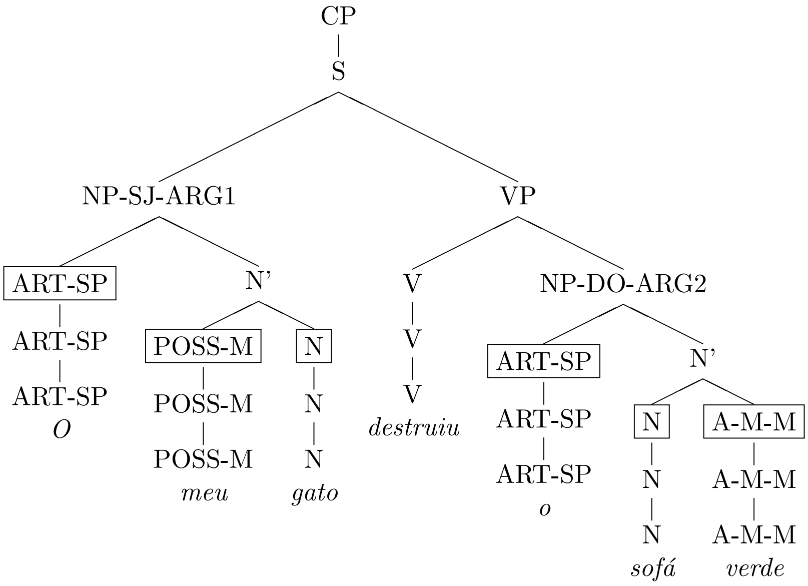


Figure 16.10: The Sg-Nominal lexical rule (corresponding node(s) in a box).

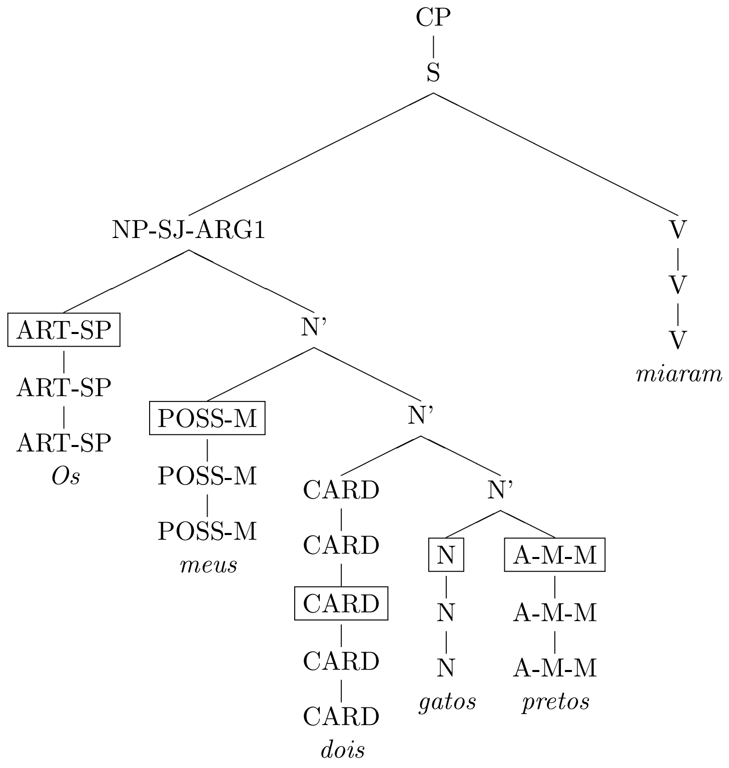


Figure 16.11: The Pl-Nominal lexical rule (corresponding node(s) in a box).

is required to be more peripheral than gender inflection, items that inflect for gender but do not possess a singular form (some cardinals, like "dois"/ "duas" in the example below) must nevertheless undergo this rule as well. Interestingly, they show the `-s` suffix typical of plural nominals, and thus, as far as this rule is concerned, are not irregular.

See Figure 16.11.

Ex.: Os meus dois gatos pretos miaram.
the my two cats black meowed
My two black cats meowed.

- Superlative-Adjective

  This rule produces the `-íssimo` forms of adjectives. It applies before gender and number inflectional rules, because this suffix is less peripheral and because the `-íssimo` forms can have morphological properties different from the ones of its base form, as the example below illustrates (where the base "contente" shows no gender agreement but the resulting form does).

  See Figure 16.12.

  Ex.: A Maria está contentíssima.
  the Maria is very happy
  Maria is very happy.

- Evaluative

  This rule produces the `-inho` forms of nouns, adjectives and past participles. It applies before gender and number inflectional rules, because this suffix is less peripheral.

  See Figure 16.13.

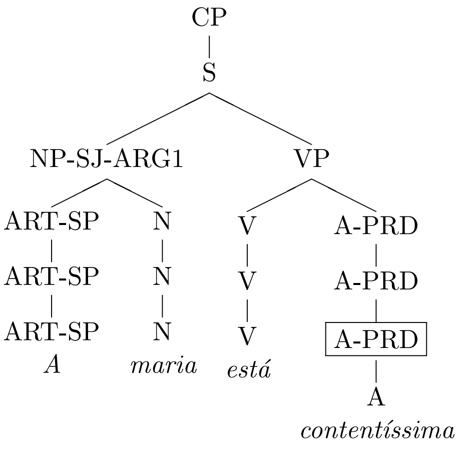


Figure 16.12: The Superlative-Adjective lexical rule (corresponding node(s) in a box).

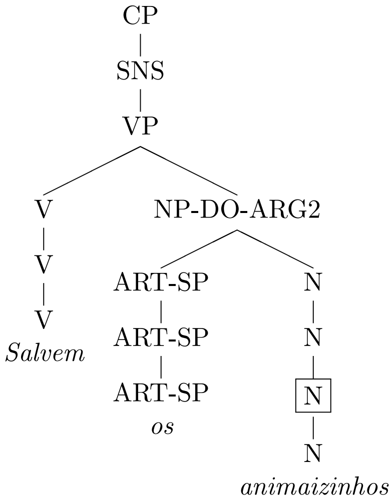


Figure 16.13: The Evaluative lexical rule (corresponding node(s) in a box).

Ex.: Salvem os animaizinhos.
save the poor animals
Save the poor animals.

16.3.1 Postponed Coverage or Known Limitations

The list of irregular forms of nouns and adjectives is not exhaustive, since LXGram is intended to be integrated with an external morphological component that contains such a list.

16.4 Punctuation

There are also a few punctuation rules in LXGram (see Section 5.9). They are more peripheral than any of the inflectional rules. They are defined in the file `punctuation.tdl`.

- Punctuation-Period
  This rule has the constraints:

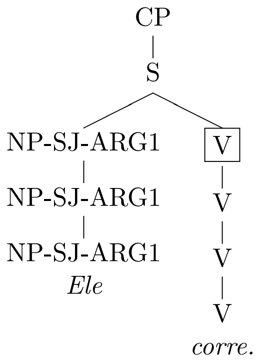


Figure 16.14: The Punctuation-Period lexical rule (corresponding node(s) in a box).

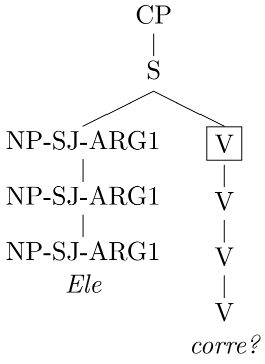


Figure 16.15: The Punctuation-Question-Mark lexical rule (corresponding node(s) in a box).

$$\begin{bmatrix} \textsc{punct}|\textsc{right}|\textsc{outer-punct} & \begin{bmatrix} \textit{period} & \\ \textsc{s-force} & \textit{proposition-or-command} \\ \textsc{elliptical} & \text{-} \end{bmatrix} \\ \textsc{dtr}|\textsc{punct}|\textsc{right}|\textsc{outer-punct} & \textit{no-punctuation-mark} \end{bmatrix}$$

See Figure 16.14.

Ex.: Ele corre.
he runs
He runs.

- Punctuation-Question-Mark
This rule has the constraints:

$$\begin{bmatrix} \textsc{punct}|\textsc{right}|\textsc{outer-punct} & \begin{bmatrix} \textit{question-mark} & \\ \textsc{s-force} & \textit{question} \\ \textsc{elliptical} & \text{-} \end{bmatrix} \\ \textsc{dtr}|\textsc{punct}|\textsc{right}|\textsc{outer-punct} & \textit{no-punctuation-mark} \end{bmatrix}$$

See Figure 16.15.

Ex.: Ele corre?
he runs
Does he run?

- Punctuation-Ellipsis
This rule has the constraints:

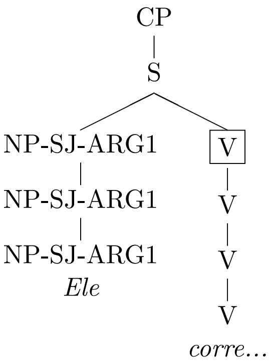


Figure 16.16: The PUNCTUATION-ELLIPSIS lexical rule (corresponding node(s) in a box).

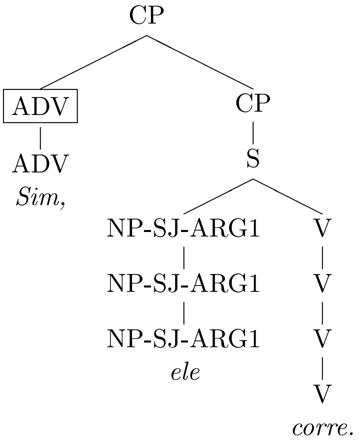


Figure 16.17: The PUNCTUATION-COMMA lexical rule (corresponding node(s) in a box).

$$\begin{bmatrix} \text{PUNCT|RIGHT|OUTER-PUNCT} & \begin{bmatrix} \textit{ellipsis} & \\ \text{S-FORCE} & \textit{proposition-or-command} \\ \text{ELLIPTICAL} & + \end{bmatrix} \\ \text{DTR|PUNCT|RIGHT|OUTER-PUNCT} & \textit{no-punctuation-mark} \end{bmatrix}$$

See Figure 16.16.

Ex.: Ele corre...
he runs
He runs...

- PUNCTUATION-COMMA
This rule has the constraints:

$$\begin{bmatrix} \text{PUNCT|RIGHT|OUTER-PUNCT} & \textit{comma} \\ \text{DTR|PUNCT|RIGHT|OUTER-PUNCT} & \textit{no-punctuation-mark} \end{bmatrix}$$

See Figure 16.17.

Ex.: Sim, ele corre.
yes he runs
Yes, he runs.

There are additional punctuation rules for: colons, semicolons, exclamation marks, left parenthesis, right parenthesis, left quotes, right quotes, left square brackets and right square brackets.

16.5 Alternations

The remaining lexical rules do not have an impact on the orthographic representation of words. They are responsible for alternations in grammatical properties of words. We present a few examples below.

- Unaccusative
  Consider the following contrast:

(209) a. A mãe desmaiou.
the mother fainted
The mother fainted.

b. A mãe está desmaiada.
The mother is fainted
The mother has fainted and is lying (on the floor, presumably).

(210) a. A mãe sorriu.
the mother smiled
The mother smiled.

b. * A mãe está sorrida.
The mother is smiled

All these sentences contain one of two verbs (*desmaiar* "faint" or *sorrir* "smile") combining with a subject and no complement. But these two verbs behave differently in their passive participle form: *desmaiar* "faint" admits it, but *sorrir* "smile" does not. In this respect, *desmaiar* "faint" is similar to transitive verbs, like *danificar* "damage":

(211) a. A criança danificou o piano.
the child damaged the piano
The child damaged the piano.

b. O piano está danificado.
the piano is damaged
The piano is damaged.

But even here, there is one significant difference. The subject of a transitive verb in a passive form, as in (211b), corresponds to the object of the same verb when it occurs in the active voice, as in (211a). But the sole argument of a verb like *desmaiar* "faint" is always its subject, regardless of whether the verb occurs in an active form or in its passive participle form.

Unaccusative verbs, like *desmaiar* "faint", are treated in LXGram in a way similar to the way they are analyzed in the GB literature. Namely they are analyzed as coming in the lexicon as selecting for no subject and a single direct object that however can never surface as a direct object. That is, the lexical entry, as it is, can never be directly combined with other words. Instead, it must either undergo the rule for passive participles (which just like for transitive verbs, just promotes the direct object to subject and removes it from the comps list) or undergo a dedicated lexical rule, named *unaccusative*, which does exactly the same thing.

- Causative-alternation
  The following sentences illustrate causative alternation:

(212) a. O navio afundou.
the ship sank
The ship sank.

b. Os piratas afundaram o navio.
the pirates sank the ship
The pirates sank the ship.

According to either sentence, the entity that goes under water is the ship. But *o navio/the ship* is the subject of the verb in the first sentence and the object of the verb in the second sentence. Because of this similarity in meaning, LXGram chooses to produce representations for these sentences where *the ship* is always the same argument of the *sinking* relation. The key to this is to assign the second sentence a representation similar to the representation for a sentence like the following one:

(213) Os piratas fizeram o navio afundar.
the pirates made the ship sink
The pirates made the ship sink.

In other words, the *sinking* relation always has a single argument. The intransitive version of the verb contains a single relation in its semantics; it is a unary relation, and its argument is the subject of the sentence. The MRS representation for the sentence in (212a) is the following:

$$\begin{bmatrix} \textit{mrs} & \\ \text{LTOP} & \boxed{h1}\ h \\ \text{INDEX} & \boxed{e2}\ e \\ \text{RELS} & \left\langle \begin{array}{l} \begin{bmatrix} \textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_navio\_n\_1\_rel} & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \end{bmatrix}, \begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h8}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{t9}\ t \end{bmatrix}, \begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h8} \\ \text{ARG0} & \boxed{t9} \\ \text{ARG1} & \boxed{now10}\ now \end{bmatrix}, \\ \begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h8} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{e12}\ e \\ \text{ARG2} & \boxed{h11}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_afundar-se\_v\_1\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{e12} \\ \text{ARG1} & \boxed{x4} \end{bmatrix} \end{array} \right\rangle \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h8} \end{bmatrix}, \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix} \right\rangle \end{bmatrix}$$

The *sinking* relation is the *_afundar-se_v_1_rel* relation. Its arg1 is the referential index of the subject NP *o navio* "the ship".

The transitive version of the verb comes with two relations in the semantic representation: a *sinking* relation, which is also unary, whose argument is the object of the verb, and a *cause* binary relation: its first argument is the subject of the verb, and its second argument is linked to the first relation. The MRS representation for the sentence in (212b)

$$\begin{bmatrix}
\textit{mrs} & \\
\text{LTOP} & \boxed{h1}\ h \\
\text{INDEX} & \boxed{e2}\ e \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h3}\ h \\ \text{ARG0} & \boxed{x4}\ x \\ \text{RSTR} & \boxed{h5}\ h \\ \text{BODY} & \boxed{h6}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_pirata\_n\_1\_rel} & \\ \text{LBL} & \boxed{h7}\ h \\ \text{ARG0} & \boxed{x4} \end{bmatrix},
\begin{bmatrix} \textit{at\_rel} & \\ \text{LBL} & \boxed{h8}\ h \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{t9}\ t \end{bmatrix},
\begin{bmatrix} \textit{is-before\_rel} & \\ \text{LBL} & \boxed{h8} \\ \text{ARG0} & \boxed{t9} \\ \text{ARG1} & \boxed{now10}\ now \end{bmatrix},
\begin{bmatrix} \textit{aspectual-operator\_rel} & \\ \text{LBL} & \boxed{h8} \\ \text{ARG0} & \boxed{e2} \\ \text{ARG1} & \boxed{e12}\ e \\ \text{ARG2} & \boxed{h11}\ h \end{bmatrix}, \\
\begin{bmatrix} \textit{cause\_v\_rel} & \\ \text{LBL} & \boxed{h11} \\ \text{ARG0} & \boxed{e12} \\ \text{ARG1} & \boxed{x4} \\ \text{ARG2} & \boxed{h13}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_afundar-se\_v\_1\_rel} & \\ \text{LBL} & \boxed{h14}\ h \\ \text{ARG0} & \boxed{e16}\ e \\ \text{ARG1} & \boxed{x15}\ x \end{bmatrix},
\begin{bmatrix} \textit{\_o\_q\_rel} & \\ \text{LBL} & \boxed{h17}\ h \\ \text{ARG0} & \boxed{x15} \\ \text{RSTR} & \boxed{h18}\ h \\ \text{BODY} & \boxed{h19}\ h \end{bmatrix},
\begin{bmatrix} \textit{\_navio\_n\_1\_rel} & \\ \text{LBL} & \boxed{h20}\ h \\ \text{ARG0} & \boxed{x15} \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \left\langle
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h1} \\ \text{LARG} & \boxed{h8} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h5} \\ \text{LARG} & \boxed{h7} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h13} \\ \text{LARG} & \boxed{h14} \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{h18} \\ \text{LARG} & \boxed{h20} \end{bmatrix}
\right\rangle
\end{bmatrix}$$

The same *_afundar-se_v_1_rel* relation (the *sinking* relation) similarly has as its ARG1 the referential index corresponding to the NP *o navio* “the ship”, even though it is the direct object of the verb in this sentence. There is an additional *cause_v_rel* relation. Its ARG1 is the referential index of the subject NP *os piratas* “the pirates”. Its ARG2 is related with the *_afundar-se_v_1_rel* relation via a *qeq* constraint.

In LXGram, this alternation is accounted for by means of a lexical rule which produces the transitive version from the intransitive one. This rule adds the *cause_v_rel* relation to the MRS representation and changes the verb’s valence: the subject becomes the direct object (but still linked to the ARG1 of the verb’s base relation) and another subject is added as the ARG1 of the new *cause_v_rel* relation.

Chapter 17

Test Suite

This chapter describes the test suites used in LXGram.

The test suites attempt to attest many combinations of phenomena and word patterns systematically. Figure 17.1 shows an example of this. This makes it easier to debug if needed, since it allows one to understand better where the problem is. It also enables one to check more thoroughly whether changes in one place in the grammar have made damage in other analyses.

```
Ele mora perto do centro.
Ele mora perto.
Ele mora muito perto do centro.
Ele mora muito perto.
Ele mora mais perto do centro.
Ele mora mais perto.
Ele mora muito mais perto do centro.
Ele mora muito mais perto.
```

Figure 17.1: Example test suite sentences

In the test suite for the phase A.1 (see Chapter 4), the test items are grouped by phenomenon. Additionally, each test item is tagged with several pieces of information, as depicted in Figure 17.2.

The fields that are filled in for each test item are the following:

- TestItem
  The test sentence, with additional | symbols for purposes of alignment with the gloss

- Gloss
  A gloss in English

- English
  The English translation

- Tagged
  The test sentence with part of speech tags as well as inflectional tags. These tags are produced manually.

- OtherPhenomena
  Other phenomena that the test item attests, besides the main phenomena under the section of which the test example appears

- TSNLPhenomenon
  Classification of this test item according to the phenomena list presented in [Oepen et al., 1997] and used in [incr tsdb()].

Some figures on the test suites developed so far as well as on grammar coverage and efficiency when parsing them can be found in Chapter 4.

Currently, the test suite for the phase A.1 is annotated and contains 202 items. The test suite for the phase A.4 is not annotated (it contains 851 items).

```
; @BeginPhenomenon Basic_Phrase_Structure_AdvP
...

; @TestItem   Ele | mora | perto | do | centro.
; @Gloss      he | lives | near | of the | center
; @English    He lives near the center.
; @Tagged     Ele/PRS#ms3 mora/MORAR/V/pi-3s perto/ADV de_/PREP o/DA#ms centro/CENTRO/CN#ms
; @OtherPhenomena Basic_Phrase_Structure_VP Basic_Phrase_Structure_PP Basic_Phrase_Structure
; @TSNLPPhenomenon  C_Complementation
Ele mora perto do centro.
```

Figure 17.2: Example test suite item with annotations

17.1 Ontology of Phenomena

Besides the classification of phenomena presented in [Oepen et al., 1997], which is roughly the one available in [incr tsdb()], test items are also being classified using a different ontology of phenomena, using the tags BeginPhenomenon and OtherPhenomena presented above.

This ontology follows the implementation agenda defined in Section 2.2 and is expanded as progress is made. Currently the ontology is the following:

- Basic_Phrase_Structure_VP
- Basic_Phrase_Structure_PP
- Basic_Phrase_Structure_AP
- Basic_Phrase_Structure_AdvP
- Auxiliaries
- Basic_Phrase_Structure_NP

Chapter 18

Appendix I - Lexical Types

This appendix contains the exhaustive list of the types that are used in LXGram for lexical entries.

- *adjective-isect-comparative-lex*
  Exs.: “menor”.

- *adjective-isect-comparative-postnom-lex*
  Exs.: “maior”, “melhor”, “pior”.

- *adjective-isect-nonpred-gradable-prepostnom-gender_infl-1comp_a-lex*
  Exs.: “respectivo”.

- *adjective-isect-nonpred-ungradable-postnom-gender_infl-no_plural-0comps-lex*
  Exs.: “tantos”.

- *adjective-isect-nonpred-ungradable-postnom-gender_uninfl-no_plural-0comps-lex*
  Exs.: “tal”.

- *adjective-isect-subj_np-gradable-postnom-gender_infl-0comps-lex*
  Exs.: “tchecoslovaco”, “vago”, “beirão”, “filarmônico”, “negro”.

- *adjective-isect-subj_np-gradable-postnom-gender_infl-1comp_a-lex*
  Exs.: “superior”.

- *adjective-isect-subj_np-gradable-postnom-gender_infl-1comp_com-lex*
  Exs.: “atencioso”.

- *adjective-isect-subj_np-gradable-postnom-gender_uninfl-0comps-lex*
  Exs.: “estudantil”, “colecionável”, “estadunidense”, “otimista”, “israelense”.

- *adjective-isect-subj_np-gradable-postnom-gender_uninfl-1comp_com-lex*
  Exs.: “coincidente”.

- *adjective-isect-subj_np-gradable-postnom-gender_uninfl-1comp_de-lex*
  Exs.: “diferente”.

- *adjective-isect-subj_np-gradable-postnom-gender_uninfl-1comp_em-lex*
  Exs.: “residente”.

- *adjective-isect-subj_np-gradable-postnom-synth-gender_infl-0comps-lex*
  Exs.: “bom”, “mau”.

- *adjective-isect-subj_np-gradable-postnom-synth-gender_uninfl-0comps-lex*
  Exs.: “grande”.

- *adjective-isect-subj_np-gradable-postnom-uninfl-0comps-lex*
  Exs.: “cor-de-rosa”, “simples”, “magenta”, “laranja”, “cor-de-laranja”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-0comps-lex*
  Exs.: “direto”, “escuro”, “ranhoso”, “cómodo”, “barato”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_a-lex*
  Exs.: “idêntico”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_a_com-lex*
  Exs.: “parecido”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_com-lex*
  Exs.: “furioso”, “satisfeito”, “sincero”, “tranquilo”, “tranqüilo”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_de-lex*
  Exs.: “homónimo”, “homônimo”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_em-lex*
  Exs.: “pioneiro”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_infl-1comp_para-lex*
  Exs.: “simpático”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: “careta”, “omnipotente”, “frequente”, “quente”, “breve”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_uninfl-1comp_a-lex*
  Exs.: “anterior”, “igual”, “semelhante”, “susceptível”, “suscetível”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_uninfl-1comp_com-lex*
  Exs.: “contente”, “feliz”, “infeliz”.

- *adjective-isect-subj_np-gradable-prepostnom-gender_uninfl-1comp_para-lex*
  Exs.: “disponível”.

- *adjective-isect-subj_np-gradable-prepostnom-uninfl-0comps-lex*
  Exs.: “reles”.

- *adjective-isect-subj_np-ungradable-postnom-gender_infl-0comps-lex*
  Exs.: “alfabético”.

- *adjective-isect-subj_np-ungradable-postnom-gender_uninfl-0comps-lex*
  Exs.: “alimentar”.

- *adjective-isect-subj_np-ungradable-prepostnom-gender_infl-0comps-lex*
  Exs.: “museológico”.

- *adjective-isect-subj_np-ungradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: “existente”.

- *adjective-isect-subj_np_cp+conj-gradable-postnom-gender_infl-1comp_a-lex*
  Exs.: “relativo”.

- *adjective-isect-subj_np_cp+ind-gradable-postnom-gender_infl-0comps-lex*
  Exs.: “óbvio”.
- *adjective-isect-subj_np_cp+ind-gradable-postnom-gender_uninfl-0comps-lex*
  Exs.: “evidente”.
- *adjective-isect-subj_np_inf-gradable-postnom-gender_infl-0comps-lex*
  Exs.: “provocatório”.
- *adjective-isect-subj_np_inf-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: “habitual”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_infl-0comps-lex*
  Exs.: “abjecto”, “abjeto”, “controverso”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_infl-1comp_pp_de_para_slash-lex*
  Exs.: “desnecessário”, “válido”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_uninfl-0comps-lex*
  Exs.: “aconselhável”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_uninfl-1comp_de-lex*
  Exs.: “esclarecedor”, “protector”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_uninfl-1comp_por-lex*
  Exs.: “aceitável”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-postnom-gender_uninfl-1comp_pp_de_slash-lex*
  Exs.: “difícil”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-prepostnom-gender_infl-0comps-lex*
  Exs.: “asqueroso”.
- *adjective-isect-subj_np_inf_cp+conj-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: “aterrador”.
- *adjective-noun_argument-gender_infl-lex*
  Exs.: “bolonhês”, “coreano”, “escalabitano”, “argentino”, “português”.
- *adjective-noun_argument-gender_uninfl-lex*
  Exs.: “israelita”, “carioca”, “esquimó”, “israelense”, “moscovita”.
- *adjective-qualquer-lex*
  Exs.: “qualquer”.
- *adjective-scopal-comparative-lex*
  Exs.: “melhor”, “pior”.
- *adjective-scopal-comparative-nonpred-prenom-lex*
  Exs.: “maior”.
- *adjective-scopal-nonpred-gradable-prenom-gender_infl-0comps-lex*
  Exs.: “alto”, “belo”.
- *adjective-scopal-nonpred-gradable-prenom-synth-gender_uninfl-0comps-lex*
  Exs.: “grande”.

- *adjective-scopal-nonpred-ungradable-prenom-gender_infl-0comps-lex*
  Exs.: "futuro", "mero", "rico".
- *adjective-scopal-nonpred-ungradable-prenom-gender_uninfl-0comps-lex*
  Exs.: "pobre".
- *adjective-scopal-nonpred-ungradable-prenom-uninfl-0comps-lex*
  Exs.: "simples".
- *adjective-scopal-subj_np-pred_exist_sem-gradable-prepostnom-gender_infl-0comps-lex*
  Exs.: "incerto".
- *adjective-scopal-subj_np-pred_exist_sem-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: "possível", "provável".
- *adjective-scopal-subj_np-pred_subj_n_sem-gradable-postnom-gender_infl-0comps-lex*
  Exs.: "suspenso".
- *adjective-scopal-subj_np-pred_subj_n_sem-gradable-prepostnom-gender_infl-0comps-lex*
  Exs.: "correcto", "máximo", "antigo", "avançado", "falso".
- *adjective-scopal-subj_np-pred_subj_n_sem-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: "competente", "incompetente", "triste".
- *adjective-scopal-subj_np-pred_subj_n_sem-gradable-prepostnom-gender_uninfl-1comp_para-lex*
  Exs.: "importante".
- *adjective-scopal-subj_np-pred_subj_n_sem-gradable-prepostnom-synth-gender_infl-0comps-lex*
  Exs.: "bom", "mau".
- *adjective-scopal-subj_np_cp+ind_cp+conj-pred_exist_sem-gradable-prepostnom-gender_uninfl-1comp_por-lex*
  Exs.: "inquestionável".
- *adjective-scopal-subj_np_inf_cp+conj-pred_exist_sem-gradable-prepostnom-gender_uninfl-0comps-lex*
  Exs.: "inevitável".
- *adjective-scopal-subj_np_inf_cp+conj-pred_subj_n_sem-gradable-prepostnom-gender_infl-0comps-lex*
  Exs.: "genuíno".
- *adjective-scopal-subj_np_inf_cp+conj-pred_subj_n_sem-gradable-prepostnom-gender_uninfl-1comp_de_em-lex*
  Exs.: "habitual".
- *adjective-superlative-lex*
  Exs.: "óptimo", "ótimo".
- *adverb-degree-1comp_como_quanto-lex*
  Exs.: "tão".
- *adverb-degree-gradable-lex*
  Exs.: "meio", "quase", "assaz", "algo", "buereré".

- *adverb-degree-mais_do_que-lex*
  Exs.: “chus”, “mais”.

- *adverb-degree-menos_do_que-lex*
  Exs.: “menos”.

- *adverb-degree-ungradable-infl-masc-or-fem-lex*
  Exs.: “todo”.

- *adverb-degree-ungradable-infl-neut-lex*
  Exs.: “tudo”.

- *adverb-degree-ungradable-lex*
  Exs.: “extremamente”, “tão”.

- *adverb-degree_of_comparison-lex*
  Exs.: “ainda”.

- *adverb-habitual-lex*
  Exs.: “habitualmente”.

- *adverb-habitual-optional_proctrigger-lex*
  Exs.: “diariamente”.

- *adverb-interrogative-in_situ-lex*
  Exs.: “porquê”, “quando”, “como”.

- *adverb-interrogative-lex*
  Exs.: “porque”, “quando”, “como”.

- *adverb-interrogative-locative-in_situ-lex*
  Exs.: “onde”, “aonde”.

- *adverb-interrogative-locative-lex*
  Exs.: “onde”, “aonde”.

- *adverb-interrogative-locative_complement-in_situ-lex*
  Exs.: “onde”, “aonde”.

- *adverb-interrogative-locative_complement-lex*
  Exs.: “onde”, “aonde”.

- *adverb-isect-gradable-no_subj-0comps-lex*
  Exs.: “de quando em vez”, “em verdade”, “a as escâncaras”, “todavia”, “a_ _a surdina”.

- *adverb-isect-gradable-no_subj-0comps-modifying-verbs-adjectives-or-adverbs-lex*
  Exs.: “de cor”, “cabonde”, “aldemenos”, “ao de leve”, “alto e bom som”.

- *adverb-isect-gradable-optional_proctrigger-no_subj-0comps-modifying-verbs-lex*
  Exs.: “espontaneamente”, “humildemente”.

- *adverb-isect-gradable-proctrigger-no_subj-0comps-modifying-verbs-lex*
  Exs.: “pouco a pouco”, “carinhosamente”, “aginha”, “muito”, “depressa”.

- *adverb-isect-subj-0comps-lex*
  Exs.: “a_ _os tombos”, “a desoras”, “a cotio”, “a a sorte”, “sem tir-te nem guar-te”.

- *adverb-isect-subj-0comps-postverbal-lex*
  Exs.: “bem”, “mal”.

- *adverb-isect-subj-0comps-proctrigger-lex*
  Exs.: “assi”, “assim”, “debalde”, “em balde”, “em vão”.

- *adverb-isect-subj-1comp_com-lex*
  Exs.: “a compasso”, “a par e passo”.

- *adverb-isect-subj-1comp_de-lex*
  Exs.: “aparte”, “a tempo”, “a_ _as avessas”, “a par”, “a passo”.

- *adverb-isect-ungradable-no_subj-0comps-lex*
  Exs.: “emmentos”, “para cúmulo”, “hoje em dia”, “sobretal”, “em suma”.

- *adverb-isect-ungradable-no_subj-0comps-modifying-verbs-adjectives-or-adverbs-lex*
  Exs.: “em aquele entretanto”, “por o sim por o não”, “em o interim”, “entretanto”, “rebém”.

- *adverb-isect-ungradable-no_subj-0comps-modifying-verbs-lex*
  Exs.: “contudo”, “em especial”.

- *adverb-isect-ungradable-postnom-uninfl-1comp_s_or_como_s-lex*
  Exs.: “tal e qual”.

- *adverb-isect-ungradable-proctrigger-no_subj-0comps-modifying-verbs-lex*
  Exs.: “outrossim”, “item a item”, “directamente”, “item por item”, “em um ápice”.

- *adverb-isect-ungradable-subj-0comps-lex*
  Exs.: “em _a mesma”, “em a mesma”.

- *adverb-locative-0comps-lex*
  Exs.: “aqui”, “alhures”, “avante”, “algures”, “adiante”.

- *adverb-locative-1comp_a_de-lex*
  Exs.: “defronte”, “junto”.

- *adverb-locative-1comp_de-lex*
  Exs.: “a a desbanda”, “a_ _o de cima”, “embaixo”, “a a desamão”, “dentro”.

- *adverb-locative-or-temporal-subj-1comp-lex*
  Exs.: “antes”, “depois”.

- *adverb-locative_complement-0comps-lex*
  Exs.: “a sestro”, “aló”, “aí”, “algures”, “alhures”.

- *adverb-locative_complement-1comp_a_de-lex*
  Exs.: “defronte”, “fora”, “junto”.

- *adverb-locative_complement-1comp_de-lex*
  Exs.: “a o de cima”, “trás”, “dentro”, “além”, “longe”.

- *adverb-np-adjunct-lex*
  Exs.: “tão-pouco”, “tão-só”, “apenas”, “tampouco”, “somente”.

- *adverb-np-adjunct-npi-lex*
  Exs.: “sequer”.
- *adverb-np-adjunct-prehead-lex*
  Exs.: “até mesmo”, “mesmo”.
- *adverb-np-adjunct-prehead-n_word-lex*
  Exs.: “nem”.
- *adverb-np-adjunct-prehead-npi-lex*
  Exs.: “nem”.
- *adverb-relative-nonpredicational-locative-lex*
  Exs.: “onde”, “aonde”.
- *adverb-relative-predicational-lex*
  Exs.: “onde”, “aonde”, “como”.
- *adverb-scopal-gradable-no_subj-0comps-lex*
  Exs.: “adur”, “provavelmente”, “comummente”, “tipicamente”, “de ordinário”.
- *adverb-scopal-gradable-proctrigger-no_subj-0comps-modifying-verbs-adjectives-or-adverbs-lex*
  Exs.: “samica”, “ainda”, “frequentemente”, “sempre”, “freqüentemente”.
- *adverb-scopal-pred_modifier-lex*
  Exs.: “tampouco”, “tão-só”, “meramente”, “só”, “até mesmo”.
- *adverb-scopal-pred_modifier-npi-lex*
  Exs.: “sequer”.
- *adverb-scopal-pred_modifier-nword-lex*
  Exs.: “nem sequer”.
- *adverb-scopal-pred_modifier-prehead-n_word-lex*
  Exs.: “nem”.
- *adverb-scopal-proctrigger-preverbal-lex*
  Exs.: “a modos que”, “bem”, “mal”.
- *adverb-scopal-temporal-proctrigger-n_word-lex*
  Exs.: “jamais”, “nanja”, “nunca”, “nunca por nunca”.
- *adverb-scopal-temporal-proctrigger-npi-lex*
  Exs.: “jamais”, “nanja”, “nunca”, “nunca por nunca”.
- *adverb-scopal-ungradable-no_subj-0comps-lex*
  Exs.: “alegadamente”.
- *adverb-scopal-ungradable-no_subj-1comp_a-lex*
  Exs.: “comparativamente”.
- *adverb-temporal-future-lex*
  Exs.: “amanhã”, “crás”.
- *adverb-temporal-lex*
  Exs.: “antestempo”, “antessazão”, “sonoite”, “agora”, “outrora”.

- *adverb-temporal-past-lex*
  Exs.: “trás-anteontem”, “trasantontem”, “tresantontem”, “ontem”, “antanho”.
- *adverb-temporal-proctrigger-lex*
  Exs.: “breve”, “tarde”, “logo”, “cedinho”, “já”.
- *cardinal-2digit-lex*
  Exs.: “cinqüenta”, “sessenta”, “oitenta”, “setenta”, “trinta”.
- *cardinal-3digit-gender_infl-lex*
  Exs.: “seiscentos”, “novecentos”, “duzentos”, “quinhentos”, “trezentos”.
- *cardinal-3digit-gender_uninfl-lex*
  Exs.: “cento”.
- *cardinal-4digit-lex*
  Exs.: “mil”.
- *cardinal-atomic-gender_infl-pl-lex*
  Exs.: “dois”.
- *cardinal-atomic-gender_uninfl-pl-lex*
  Exs.: “doze”, “quatro”, “m”, “cem”, “seis”.
- *cardinal-atomic-sg-lex*
  Exs.: “um”.
- *comparative-particle-como_quanto-lex*
  Exs.: “como”, “quanto”.
- *comparative-particle-do_que-lex*
  Exs.: “de _o que”, “de o que”, “que”.
- *complementizer-decl-com+que-lex*
  Exs.: “com que”.
- *complementizer-decl-lex*
  Exs.: “que”.
- *complementizer-interr-lex*
  Exs.: “se”.
- *complementizer-interr-wh-lex*
  Exs.: “é que”, “que”.
- *conjunction-events_only-lex*
  Exs.: “mas”.
- *conjunction-lex*
  Exs.: “assim como”, “bem como”, “e”, “ou”.
- *conjunction-n_word-lex*
  Exs.: “nem”.
- *conjunction-npi-lex*
  Exs.: “nem”.

- *copular-verb-predicational-inherent_clitic-lex*
  Exs.: “apresentar”, “encontrar”.

- *copular-verb-predicational-lex*
  Exs.: “continuar”, “ficar”, “permanecer”.

- *definite-article-lex*
  Exs.: “o”, “_o”.

- *definite-article-nosem-lex*
  Exs.: “o”, “_o”.

- *demonstrative-masc-or-fem-lex*
  Exs.: “este”, “esse”, “aquele”.

- *demonstrative-neut-lex*
  Exs.: “isto”, “isso”, “aquilo”.

- *demonstrative-noquant-masc-or-fem-lex*
  Exs.: “este”, “esse”, “aquele”.

- *demonstrative-noquant-neut-lex*
  Exs.: “isto”, “isso”, “aquilo”.

- *demonstrative-postnominal-masc_with_fem-lex*
  Exs.: “este”, “esse”, “aquele”.

- *determiner-indef-empty_lp-gender_uninfl_non_neuter-multi_um-lex*
  Exs.: “qualquer”.

- *determiner-indef-empty_lp-masc_with_fem-lex*
  Exs.: “cada um”.

- *determiner-indef-full_lp-uninfl-lex*
  Exs.: “cada”.

- *determiner-indef-gender_uninfl-lex*
  Exs.: “bastante”.

- *determiner-indef-gender_uninfl_non_neuter-lex*
  Exs.: “qualquer”.

- *determiner-indef-masc-sg-null_noun-lex*
  Exs.: “alguém”.

- *determiner-indef-masc-sg-null_noun-n_word-lex*
  Exs.: “ninguém”.

- *determiner-indef-masc-sg-null_noun-npi-lex*
  Exs.: “ninguém”.

- *determiner-indef-masc_or_fem-lex*
  Exs.: “algum”, “quase nenhum”.

- *determiner-indef-masc_or_fem-n_word-lex*
  Exs.: “nenhum”.

- *determiner-indef-masc_or_fem-npi-lex*
  Exs.: “nenhum”.

- *determiner-indef-mascneut_or_fem-1comp_como_quanto-lex*
  Exs.: “tanto”.

- *determiner-indef-mascneut_or_fem-lex*
  Exs.: “tanto”, “um”.

- *determiner-indef-mascneut_or_fem-multi_mais-lex*
  Exs.: “muito”, “pouco”.

- *determiner-indef-mascneut_or_fem-multi_qualquer-lex*
  Exs.: “um”.

- *determiner-indef-mascneut_or_fem-pl-multi_e_tantos-lex*
  Exs.: “tanto”.

- *determiner-indef-mascneut_or_fem-pl-multi_menos-lex*
  Exs.: “muito”, “pouco”.

- *determiner-indef-mascneut_or_fem-pl-multi_poucos-lex*
  Exs.: “um”.

- *determiner-indef-mascneut_or_fem-pl-multi_quantos-lex*
  Exs.: “um”.

- *determiner-indef-mascneut_or_fem-pl-multi_tantos-lex*
  Exs.: “outro”, “um”.

- *determiner-indef-neut-lex*
  Exs.: “algo”, “quase nada”.

- *determiner-indef-neut-n_word-lex*
  Exs.: “nada”.

- *determiner-indef-neut-npi-lex*
  Exs.: “nada”.

- *determiner-indef-no_noun_ellipsis-lex*
  Exs.: “só”.

- *determiner-indef-specific-masc_or_fem-lex*
  Exs.: “certo”, “determinado”.

- *determiner-indef-uninfl-lex*
  Exs.: “pouco menos”, “pouco mais”, “muito menos”, “muito mais”, “menos”.

- *determiner-particle-e_tantos-lex*
  Exs.: “e”.

- *determiner-particle-mais-lex*
  Exs.: “mais”.

- *determiner-particle-menos-lex*
  Exs.: “menos”.

- *determiner-particle-poucos-lex*
  Exs.: “pouco”.

- *determiner-particle-qualquer-lex*
  Exs.: “qualquer”.

- *determiner-particle-quantos-lex*
  Exs.: “quanto”.

- *determiner-particle-tantos-lex*
  Exs.: “tanto”.

- *determiner-particle-um-lex*
  Exs.: “um”.

- *discontinuous-conjunction-lex*
  Exs.: “mas também”, “como também”, “seja”, “quer”, “como ainda”.

- *discontinuous-conjunction-n_word-lex*
  Exs.: “nem”.

- *discontinuous-conjunction-no_semantics-lex*
  Exs.: “não só”, “ora”, “ou”, “seja”, “quer”.

- *discontinuous-conjunction-no_semantics-n_word-lex*
  Exs.: “nem”.

- *discontinuous-conjunction-no_semantics-npi-lex*
  Exs.: “nem”.

- *discontinuous-conjunction-npi-lex*
  Exs.: “nem”.

- *discourse-element-lex*
  Exs.: “ena”, “ah”, “cruzes”, “jesus cristo”, “prontos”.

- *indefinite-specific-lex*
  Exs.: “certo”, “determinado”.

- *interrogative-determiner-gender_infl-lex*
  Exs.: “quanto”.

- *interrogative-determiner-gender_uninfl-lex*
  Exs.: “qual”.

- *interrogative-determiner-in_situ-gender_infl-lex*
  Exs.: “quanto”.

- *interrogative-determiner-in_situ-gender_uninfl-lex*
  Exs.: “qual”.

- *interrogative-determiner-in_situ-que-lex*
  Exs.: “que”.

- *interrogative-determiner-que-lex*
  Exs.: “que”.

- *interrogative-pronoun-in_situ-o_que-lex*
  Exs.: “o quê”, “quê”.

- *interrogative-pronoun-in_situ-quem-lex*
  Exs.: “quem”.

- *interrogative-pronoun-o_que-lex*
  Exs.: “o que”, “que”.

- *interrogative-pronoun-quem-lex*
  Exs.: “quem”.

- *mesmo-lex*
  Exs.: “mesmo”.

- *neg-lex*
  Exs.: “não”.

- *non-numeral-ordinal-lex*
  Exs.: “enésimo”, “último”, “penúltimo”, “antepenúltimo”, “próximo”.

- *noun-common-eventive-fem-1comp_de-lex*
  Exs.: “vitória”, “atuação”, “alunagem”, “derrota”, “decisão”.

- *noun-common-fem-0comps-lex*
  Exs.: “carona”, “linguiça”, “roda”, “mulher”, “interatividade”.

- *noun-common-fem-0comps-time_unit-lex*
  Exs.: “década”, “hora”, “primavera”, “semana”.

- *noun-common-fem-1comp_de-lex*
  Exs.: “velocidade”, “secção”, “sequência”, “chegada”, “carta”.

- *noun-common-fem-2comps_de+a-lex*
  Exs.: “dádiva”.

- *noun-common-fem-2comps_de+para-lex*
  Exs.: “contribuição”.

- *noun-common-fem-2comps_de+por-lex*
  Exs.: “manifestação”.

- *noun-common-fem-2comps_de_por+de-lex*
  Exs.: “selecção”, “venda”, “fotografia”, “seleção”, “exposição”.

- *noun-common-fem-2comps_por+de-lex*
  Exs.: “correcção”, “liquidação”, “proteção”, “indenização”, “amnistia”.

- *noun-common-fem-2nd_or_3rd-1comp_de-lex*
  Exs.: “mãe”.

- *noun-common-fem-pl-0comps-lex*
  Exs.: “calças”, “nadadeiras”.

- *noun-common-fem-sg_or_pl-0comps-lex*
  Exs.: “bílis”, “miosótis”.

- *noun-common-masc-0comps-lex*
  Exs.: “grampeador”, “produto”, “moinho”, “penalti”, “cabelo”.
- *noun-common-masc-0comps-time_unit-lex*
  Exs.: “milénio”, “mês”, “segundo”, “inverno”, “outono”.
- *noun-common-masc-1comp_de-lex*
  Exs.: “vôo”, “tempo”, “objecto”, “contrário”, “jacto”.
- *noun-common-masc-2comps_de+a-lex*
  Exs.: “benefício”, “pedido”.
- *noun-common-masc-2comps_de+por-lex*
  Exs.: “interesse”.
- *noun-common-masc-2comps_de_por+de-lex*
  Exs.: “consumo”, “quadro”, “retrato”.
- *noun-common-masc-2comps_por+de-lex*
  Exs.: “armazenamento”, “registro”, “planeamento”, “batismo”, “coleccionamento”.
- *noun-common-masc-2nd_or_3rd-1comp_de-lex*
  Exs.: “pai”.
- *noun-common-masc-pl-0comps-lex*
  Exs.: “binóculos”, “brócolis”, “brócolos”.
- *noun-common-masc-pl-1comp_de-lex*
  Exs.: “meados”.
- *noun-common-masc-sg_or_pl-0comps-lex*
  Exs.: “tênis”, “húmus”, “kW”, “m/s”, “ténis”.
- *noun-common-masc_or_fem-0comps-lex*
  Exs.: “autodidata”, “manifestante”, “moscovita”, “retalhista”, “linguista”.
- *noun-common-masc_or_fem-1comp_de-lex*
  Exs.: “ativista”, “colecionista”, “activista”, “participante”, “coleccionista”.
- *noun-common-masc_or_fem-2nd_or_3rd-0comps-lex*
  Exs.: “bebé”, “polícia”, “policial”.
- *noun-common-masc_or_fem-2nd_or_3rd-1comp_de-lex*
  Exs.: “presidente”.
- *noun-common-masc_with_fem-0comps-lex*
  Exs.: “nova-iorquino”, “ator”, “paraguaio”, “indiano”, “funcionário”.
- *noun-common-masc_with_fem-1comp_de-lex*
  Exs.: “freqüentador”, “domador”, “investigador”, “inspector”, “colecionador”.
- *noun-common-masc_with_fem-2nd_or_3rd-0comps-lex*
  Exs.: “cônego”, “arquiteto”, “arquitecto”, “menino”, “cónego”.
- *noun-common-masc_with_fem-2nd_or_3rd-1comp_de-lex*
  Exs.: “director”, “diretor”, “irmão”, “ministro”, “professor”.

- *noun-common-pretitle-masc_with_fem-2nd_or_3rd-0comps-lex*
  Exs.: “senhor”.

- *noun-proper-fem-mod_det-lex*
  Exs.: “África”, “Iorque”, “Cuba”, “Cingapura”, “Mortágua”.

- *noun-proper-fem-obl_det-lex*
  Exs.: “Lapónia”, “Ártico”, “Estônia”, “Tchéquia”, “Antártida”.

- *noun-proper-fem-opt_det-2nd_or_3rd-lex*
  Exs.: “Clara”, “Francisca”, “Mariana”, “Mônica”, “Maluda”.

- *noun-proper-fem-opt_det-lex*
  Exs.: “Espanha”, “França”, “Inglaterra”.

- *noun-proper-fem-pl-obl_det-lex*
  Exs.: “Canárias”.

- *noun-proper-fem-sg_or_pl-mod_det-lex*
  Exs.: “Paris”, “Vénus”, “Vênus”.

- *noun-proper-fem-sg_or_pl-obl_det-lex*
  Exs.: “AIDS”, “Diabetes”, “Flandres”.

- *noun-proper-fem-sg_or_pl-opt_det-2nd_or_3rd-lex*
  Exs.: “Íris”.

- *noun-proper-masc-mod_det-lex*
  Exs.: “Maio”, “Neptuno”, “Netuno”, “Greensgrow”, “Moçambique”.

- *noun-proper-masc-obl_det-lex*
  Exs.: “Irã”, “Iucatão”, “Islã”, “Cancro”, “Islão”.

- *noun-proper-masc-opt_det-2nd_or_3rd-lex*
  Exs.: “Rui”, “Fidel”, “Antônio”, “João”, “Fernando”.

- *noun-proper-masc-opt_det-lex*
  Exs.: “NumberErsatz”, “Montijo”.

- *noun-proper-masc-pl-obl_det-lex*
  Exs.: “DecadeErsatz”, “Açores”, “Estados Unidos”.

- *noun-proper-masc-sg-mod_det-lex*
  Exs.: “YearOrNumberErsatz”.

- *noun-proper-masc-sg_or_pl-mod_det-lex*
  Exs.: “Zeus”.

- *noun-proper-masc-sg_or_pl-obl_det-lex*
  Exs.: “Amazonas”.

- *noun-proper-masc-sg_or_pl-opt_det-2nd_or_3rd-lex*
  Exs.: “Egas”.

- *noun-proper-masc_or_fem-mod_det-lex*
  Exs.: “Faro”.

- *noun-proper-masc_or_fem-obl_det-lex*
  Exs.: “Burger King”, “A.”.

- *noun-proper-masc_or_fem-opt_det-2nd_or_3rd-lex*
  Exs.: “Pessoa”, “Zé”, “José”, “Carvalho”, “Gomes”.

- *noun-proper-masc_or_fem-pl-obl_det-lex*
  Exs.: “Barbados”.

- *noun-proper-masc_or_fem-sg_or_pl-mod_det-lex*
  Exs.: “Bordéus”.

- *noun-proper-masc_or_fem-sg_or_pl-obl_det-lex*
  Exs.: “McDonald’s”.

- *noun-proper-masc_or_fem-sg_or_pl-opt_det-2nd_or_3rd-lex*
  Exs.: “Antunes”, “Borges”, “Eiffel”, “Jesus”.

- *o+mesmo-lex*
  Exs.: “o”.

- *o+próprio-lex*
  Exs.: “o”.

- *ordinal-lex*
  Exs.: “sexagésimo”, “sétimo”, “nonagésimo”, “quinquagésimo”, “oitavo”.

- *outro-determiner-lex*
  Exs.: “outro”.

- *outro-postdeterminer-lex*
  Exs.: “outro”.

- *personal-pronoun-ele-expletive-lex*
  Exs.: “ele”.

- *personal-pronoun-ele-lex*
  Exs.: “ele”.

- *personal-pronoun-ele_mesmo-lex*
  Exs.: “ele”.

- *personal-pronoun-ele_próprio-lex*
  Exs.: “ele”.

- *personal-pronoun-ele_todo-lex*
  Exs.: “ele”.

- *personal-pronoun-eu-lex*
  Exs.: “eu”.

- *personal-pronoun-eu_mesmo-lex*
  Exs.: “eu”.

- *personal-pronoun-eu_próprio-lex*
  Exs.: “eu”.

- *personal-pronoun-lhe-lex*
  Exs.: “lhe”.

- *personal-pronoun-lhes-lex*
  Exs.: “lhes”.

- *personal-pronoun-me-expletive-lex*
  Exs.: “me”.

- *personal-pronoun-me-non_reflexive-lex*
  Exs.: “me”.

- *personal-pronoun-migo-lex*
  Exs.: “+migo”.

- *personal-pronoun-migo_mesmo-lex*
  Exs.: “+migo”.

- *personal-pronoun-migo_próprio-lex*
  Exs.: “+migo”.

- *personal-pronoun-mim-lex*
  Exs.: “mim”.

- *personal-pronoun-mim_mesmo-lex*
  Exs.: “mim”.

- *personal-pronoun-mim_próprio-lex*
  Exs.: “mim”.

- *personal-pronoun-nos-expletive-lex*
  Exs.: “nos”.

- *personal-pronoun-nos-lex*
  Exs.: “nos”.

- *personal-pronoun-nosco-lex*
  Exs.: “+nosco”.

- *personal-pronoun-nosco_mesmos-lex*
  Exs.: “+nosco”.

- *personal-pronoun-nosco_próprios-lex*
  Exs.: “+nosco”.

- *personal-pronoun-nosco_todos-lex*
  Exs.: “+nosco”.

- *personal-pronoun-nós-lex*
  Exs.: “nós”.

- *personal-pronoun-nós_mesmos-lex*
  Exs.: “nós”.

- *personal-pronoun-nós_próprios-lex*
  Exs.: “nós”.

- *personal-pronoun-nós_todos-lex*
  Exs.: “nós”.

- *personal-pronoun-o-lex*
  Exs.: “o”, “+o”.

- *personal-pronoun-os-lex*
  Exs.: “o”, “+o”.

- *personal-pronoun-se-expletive-lex*
  Exs.: “se”.

- *personal-pronoun-se-impessoal-lex*
  Exs.: “se”.

- *personal-pronoun-se-lex*
  Exs.: “se”.

- *personal-pronoun-si-non_reflexive-lex*
  Exs.: “si”.

- *personal-pronoun-si-reflexive-lex*
  Exs.: “si”.

- *personal-pronoun-si_mesmo-lex*
  Exs.: “si”.

- *personal-pronoun-si_próprio-lex*
  Exs.: “si”.

- *personal-pronoun-sigo-non_reflexive-lex*
  Exs.: “+sigo”.

- *personal-pronoun-sigo-reflexive-lex*
  Exs.: “+sigo”.

- *personal-pronoun-sigo_mesmo-lex*
  Exs.: “+sigo mesmo”.

- *personal-pronoun-sigo_próprio-lex*
  Exs.: “+sigo próprio”.

- *personal-pronoun-te-expletive-lex*
  Exs.: “te”.

- *personal-pronoun-te-lex*
  Exs.: “te”.

- *personal-pronoun-ti-lex*
  Exs.: “ti”.

- *personal-pronoun-ti_mesmo-lex*
  Exs.: “ti”.

- *personal-pronoun-ti_próprio-lex*
  Exs.: “ti”.

- *personal-pronoun-tigo-lex*
  Exs.: "+tigo".

- *personal-pronoun-tigo_mesmo-lex*
  Exs.: "+tigo".

- *personal-pronoun-tigo_próprio-lex*
  Exs.: "+tigo".

- *personal-pronoun-tu-lex*
  Exs.: "tu".

- *personal-pronoun-tu_mesmo-lex*
  Exs.: "tu".

- *personal-pronoun-tu_próprio-lex*
  Exs.: "tu".

- *personal-pronoun-você-lex*
  Exs.: "você".

- *personal-pronoun-você_mesmo-lex*
  Exs.: "você".

- *personal-pronoun-você_próprio-lex*
  Exs.: "você".

- *personal-pronoun-vocês-lex*
  Exs.: "vocês".

- *personal-pronoun-vocês_mesmos-lex*
  Exs.: "vocês".

- *personal-pronoun-vocês_próprios-lex*
  Exs.: "vocês".

- *personal-pronoun-vocês_todos-lex*
  Exs.: "vocês".

- *personal-pronoun-vos-expletive-lex*
  Exs.: "vos".

- *personal-pronoun-vos-lex*
  Exs.: "vos".

- *personal-pronoun-vosco-lex*
  Exs.: "+vosco".

- *personal-pronoun-vosco_mesmos-lex*
  Exs.: "+vosco".

- *personal-pronoun-vosco_próprios-lex*
  Exs.: "+vosco".

- *personal-pronoun-vosco_todos-lex*
  Exs.: "+vosco".

- *personal-pronoun-vós-lex*
  Exs.: “vós”.

- *personal-pronoun-vós_mesmos-lex*
  Exs.: “vós”.

- *personal-pronoun-vós_próprios-lex*
  Exs.: “vós”.

- *personal-pronoun-vós_todos-lex*
  Exs.: “vós”.

- *plus-particle-lex*
  Exs.: “e”, “_e_”.

- *porcento-lex*
  Exs.: “

- *possessive-meu-adjunct-bp-lex*
  Exs.: “meu”.

- *possessive-meu-adjunct-lex*
  Exs.: “meu”.

- *possessive-meu-comp1-bp-lex*
  Exs.: “meu”.

- *possessive-meu-comp1-lex*
  Exs.: “meu”.

- *possessive-meu-comp2-bp-lex*
  Exs.: “meu”.

- *possessive-meu-comp2-lex*
  Exs.: “meu”.

- *possessive-meu-comp3-bp-lex*
  Exs.: “meu”.

- *possessive-meu-comp3-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-adjunct-bp-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-adjunct-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp1-bp-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp1-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp2-bp-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp2-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp3-bp-lex*
  Exs.: “meu”.

- *possessive-meu_próprio-comp3-lex*
  Exs.: “meu”.

- *possessive-nosso-adjunct-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso-adjunct-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp1-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp1-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp2-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp2-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp3-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso-comp3-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-adjunct-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-adjunct-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp1-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp1-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp2-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp2-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp3-bp-lex*
  Exs.: “nosso”.

- *possessive-nosso_próprio-comp3-lex*
  Exs.: “nosso”.

- *possessive-próprio-adjunct-bp-lex*
  Exs.: “próprio”.

- *possessive-próprio-adjunct-lex*
  Exs.: “próprio”.

- *possessive-próprio-comp1-bp-lex*
  Exs.: “próprio”.

- *possessive-próprio-comp1-lex*
  Exs.: “próprio”.

- *possessive-próprio-comp2-bp-lex*
  Exs.: “próprio”.

- *possessive-próprio-comp2-lex*
  Exs.: “próprio”.

- *possessive-seu-second_person-adjunct-bp-lex*
  Exs.: “seu”.

- *possessive-seu-second_person-adjunct-lex*
  Exs.: “seu”.

- *possessive-seu-second_person-comp1-bp-lex*
  Exs.: “seu”.

- *possessive-seu-second_person-comp1-lex*
  Exs.: “seu”.

- *possessive-seu-second_person-comp2-bp-lex*
  Exs.: “seu”.

- *possessive-seu-second_person-comp2-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-adjunct-bp-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-adjunct-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-comp1-bp-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-comp1-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-comp2-bp-lex*
  Exs.: “seu”.

- *possessive-seu-third_person-comp2-lex*
  Exs.: “seu”.

- *possessive-seu_próprio-second_person-adjunct-bp-lex*
  Exs.: “seu”.

- *possessive-seu_próprio-second_person-adjunct-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-second_person-comp1-bp-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-second_person-comp1-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-second_person-comp2-bp-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-second_person-comp2-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-adjunct-bp-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-adjunct-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-comp1-bp-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-comp1-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-comp2-bp-lex*
  Exs.: “seu”.
- *possessive-seu_próprio-third_person-comp2-lex*
  Exs.: “seu”.
- *possessive-teu-adjunct-bp-lex*
  Exs.: “teu”.
- *possessive-teu-adjunct-lex*
  Exs.: “teu”.
- *possessive-teu-comp1-bp-lex*
  Exs.: “teu”.
- *possessive-teu-comp1-lex*
  Exs.: “teu”.
- *possessive-teu-comp2-bp-lex*
  Exs.: “teu”.
- *possessive-teu-comp2-lex*
  Exs.: “teu”.
- *possessive-teu-comp3-bp-lex*
  Exs.: “teu”.
- *possessive-teu-comp3-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-adjunct-bp-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-adjunct-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp1-bp-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp1-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp2-bp-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp2-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp3-bp-lex*
  Exs.: “teu”.

- *possessive-teu_próprio-comp3-lex*
  Exs.: “teu”.

- *possessive-vosso-adjunct-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso-adjunct-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp1-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp1-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp2-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp2-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp3-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso-comp3-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-adjunct-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-adjunct-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp1-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp1-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp2-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp2-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp3-bp-lex*
  Exs.: “vosso”.

- *possessive-vosso_próprio-comp3-lex*
  Exs.: “vosso”.

- *precardinal-correlate-a-lex*
  Exs.: “a”.

- *precardinal-correlate-até-lex*
  Exs.: “até”.

- *precardinal-correlate-e-lex*
  Exs.: “e”.

- *precardinal-inflected-lex*
  Exs.: “algum”, “um”.

- *precardinal-uninflected-2comps_a-até-lex*
  Exs.: “de”.

- *precardinal-uninflected-2comps_e-lex*
  Exs.: “entre”.

- *precardinal-uninflected-lex*
  Exs.: “somente”, “apenas”, “em _o mínimo”, “precisamente”, “perto de”.

- *predeterminer-def-quant_sem-uninfl-lex*
  Exs.: “só”.

- *preposition-locative-comp_advp-lex*
  Exs.: “entre”, “para”, “por”.

- *preposition-locative-comp_np-lex*
  Exs.: “para”, “por”.

- *preposition-locative-comp_np_nom_or_obl-lex*
  Exs.: “entre”.

- *preposition-nonpredicational-comp_advp-a-lex*
  Exs.: “a”, “a_”.

- *preposition-nonpredicational-comp_advp-até-lex*
  Exs.: “até”.

- *preposition-nonpredicational-comp_advp-de-lex*
  Exs.: “de”.

- *preposition-nonpredicational-comp_advp-para-lex*
  Exs.: “para”.

- *preposition-nonpredicational-comp_np-acercade-lex*
  Exs.: “acerca de”.

- *preposition-nonpredicational-comp_np-ate-lex*
  Exs.: “até”, “até a”, “até a_”.

- *preposition-nonpredicational-comp_np-com-lex*
  Exs.: “com”, “con-”, “con_”.

- *preposition-nonpredicational-comp_np-contra-lex*
  Exs.: “contra”.

- *preposition-nonpredicational-comp_np-em-lex*
  Exs.: “em”.

- *preposition-nonpredicational-comp_np-emrelaçãoa-lex*
  Exs.: “em relação a”, “em relação a_”.

- *preposition-nonpredicational-comp_np-entre-lex*
  Exs.: “entre”.

- *preposition-nonpredicational-comp_np-paracom-lex*
  Exs.: “para com”, “para con-”, “para con_”.

- *preposition-nonpredicational-comp_np-por+parte+de-lex*
  Exs.: “por parte de”.

- *preposition-nonpredicational-comp_np-sobre-lex*
  Exs.: “sobre”.

- *preposition-nonpredicational-comp_np_or_vp-a-lex*
  Exs.: “a”, “a_”.

- *preposition-nonpredicational-comp_np_or_vp-de-lex*
  Exs.: “de”.

- *preposition-nonpredicational-comp_np_or_vp-para-lex*
  Exs.: “para”.

- *preposition-nonpredicational-comp_np_or_vp-por-lex*
  Exs.: “por”.

- *preposition-nonpredicational-comp_vp-que-lex*
  Exs.: “que”.

- *preposition-predicational-com_case-lex*
  Exs.: “com”, “con-”, “con_”, “juntamente com”, “de acordo com”.

- *preposition-predicational-comp_advp-lex*
  Exs.: “a partir de”, “até”, “de”, “desde”.

- *preposition-predicational-comp_vacuous_advp-lex*
  Exs.: “para”, “por”.

- *preposition-predicational-comp_vp-lex*
  Exs.: “de”, “para”, “por”.

- *preposition-predicational-comp_vp-n_word-lex*
  Exs.: “sem”.

- *preposition-predicational-lex*
  Exs.: “a o coberto de”, “durante”, “a o rés de”, “a coberto de”, “desde”.

- *preposition-predicational-n_word-lex*
  Exs.: “sem”.

- *preposition-predicational-nom-lex*
  Exs.: “excepto”, “exceto”, “fora”, “salvo”, “segundo”.

- *preposition-predicational_direction-lex*
  Exs.: “até”, “até a”, “até _a”, “contra”, “em direcção a”.

- *preposition-predicational_locative-lex*
  Exs.: “em torno de”, “sob”, “após”, “arredor de”, “ante”.

- *preposition-predicational_locative-nom_or_obl-lex*
  Exs.: “entre”.

- *preposition-predicational_locative_or_direction-lex*
  Exs.: “para”.

- *preposition-progressive-a-lex*
  Exs.: “a”.

- *preposition-temporal-duration-aspect_shift_culminated_process_to_culmination-lex*
  Exs.: “em”.

- *preposition-temporal-duration-aspect_shift_process_to_culmination-lex*
  Exs.: “durante”.

- *preposition-temporal-duration-aspect_shift_resulting_state_to_culmination-lex*
  Exs.: “por”.

- *pronoun-particle-mesmo-lex*
  Exs.: “mesmo”.

- *pronoun-particle-próprio-lex*
  Exs.: “próprio”.

- *próprio-prenominal-lex*
  Exs.: “próprio”.

- *que_não-outro_absent-lex*
  Exs.: “que não”.

- *que_não-outro_present-lex*
  Exs.: “que não”.

- *relative-determiner-cujo-adjunct-lex*
  Exs.: “cujo”.

- *relative-determiner-cujo-comp1-lex*
  Exs.: “cujo”.

- *relative-determiner-cujo-comp2-lex*
  Exs.: “cujo”.

- *relative-determiner-cujo-comp3-lex*
  Exs.: “cujo”.

- *relative-pronoun-det-lex*
  Exs.: “o”, “_o”.

- *relative-pronoun-free-quem-lex*
  Exs.: “quem”.

- *relative-pronoun-nonfree-quem-lex*
  Exs.: “quem”.

- *relative-pronoun-que-lex*
  Exs.: “que”.

- *relative-pronoun-restrictive-qual-lex*
  Exs.: “qual”.

- *subordinating-conjunction-conditional-lex*
  Exs.: “se”.

- *subordinating-conjunction-indicative-lex*
  Exs.: “em _a medida em que”, “já que”, “uma vez que”, “por _o que”, “tanto que”.

- *subordinating-conjunction-lex*
  Exs.: “enquanto”, “porque”, “quando”, “se bem que”, “sempre que”.

- *subordinating-conjunction-subjunctive-lex*
  Exs.: “nem que”, “de modo a que”, “depois que”, “para que”, “antes que”.

- *subordinating-conjunction-temporal-ate+que-lex*
  Exs.: “até que”.

- *tal-determiner-lex*
  Exs.: “tal”.

- *tal-postdeterminer-lex*
  Exs.: “tal”.

- *univ-quant-over-individuals-masc-or-fem-nodet-lex*
  Exs.: “ambos”, “quase todo”, “todo”.

- *univ-quant-over-individuals-masc-or-fem-pl-lex*
  Exs.: “ambos”, “quase todo”, “todo”.

- *univ-quant-over-individuals-masc-or-fem-sg-lex*
  Exs.: “quase todo”, “todo”.

- *univ-quant-over-individuals-neut-sg-lex*
  Exs.: “quase tudo”, “tudo”.

- *univ-quant-over-parts-masc-or-fem-pl-lex*
  Exs.: “quase todo”, “todo”.

- *univ-quant-over-parts-masc-or-fem-sg-lex*
  Exs.: “quase todo”, “todo”.

- *univ-quant-over-parts-neut-sg-lex*
  Exs.: “quase tudo”.

- *vague-quantifier-gender_uninfl-plural-lex*
  Exs.: “diferente”.

- *vague-quantifier-lex*
  Exs.: “muito”, “pouco”.

- *vague-quantifier-mass-lex*
  Exs.: “imenso”.

- *vague-quantifier-plural-lex*
  Exs.: “diverso”, “inúmeros”, “numeroso”, “vário”.

- *verb-0place-lex*
  Exs.: “chover”, “nevar”.

- *verb-anticausative-inherent_clitic-comp_pp_de-lex*
  Exs.: “mascarar”, “vestir”.

- *verb-anticausative-inherent_clitic-comp_pp_em-lex*
  Exs.: “organizar”.

- *verb-anticausative-inherent_clitic-comp_pp_para-lex*
  Exs.: “preparar”.

- *verb-anticausative-inherent_clitic-lex*
  Exs.: “baptizar”, “espatifar”, “batizar”, “tranquilizar”, “partir”.

- *verb-anticausative-inherent_clitic-obl_location-lex*
  Exs.: “sentar”.

- *verb-anticausative-lex*
  Exs.: “esfriar”, “humedecer”, “subir”, “esquentar”, “acordar”.

- *verb-anticausative-obl_location-lex*
  Exs.: “mergulhar”.

- *verb-anticausative-optional_inherent_clitic-comp_pp_de-lex*
  Exs.: “encher”.

- *verb-anticausative-optional_inherent_clitic-lex*
  Exs.: “passear”, “avariar”, “acabar”, “fechar”, “alimentar”.

- *verb-anticausative-subj_np_inf_cp+conj-inherent_clitic-lex*
  Exs.: “magoar”, “alegrar”, “assustar”, “prolongar”.

- *verb-anticausative-subj_np_inf_cp+conj-optional_inherent_clitic-lex*
  Exs.: “atrasar”, “pasmar”.

- *verb-comp_cp+ind_declarative-comp_indir-lex*
  Exs.: “confidenciar”.
- *verb-comp_cp-comp_indir-lex*
  Exs.: “anunciar”, “dizer”, “prometer”, “sussurrar”.
- *verb-comp_cp_com+que-lex*
  Exs.: “fazer”.
- *verb-comp_cp_declarative-lex*
  Exs.: “suspeitar”.
- *verb-comp_cp_interrogative-comp_indir-lex*
  Exs.: “perguntar”.
- *verb-comp_np_inf_cp+conj_declarative-comp_indir-lex*
  Exs.: “solicitar”.
- *verb-comp_np_inf_cp+conj_declarative-lex*
  Exs.: “aprovar”.
- *verb-comp_np_inf_cp+ind_declarative-comp_indir-lex*
  Exs.: “avançar”, “jurar”, “referir”.
- *verb-comp_np_inf_cp+ind_declarative-lex*
  Exs.: “alegar”, “concluir”, “registar”, “registrar”, “publicar”.
- *verb-comp_np_inf_cp_declarative-lex*
  Exs.: “admitir”.
- *verb-comp_np_or_pp_com-lex*
  Exs.: “contactar”.
- *verb-comp_np_or_pp_sobre-lex*
  Exs.: “especular”.
- *verb-compound_tense_aux-lex*
  Exs.: “ter”.
- *verb-compound_tense_aux_haver-lex*
  Exs.: “haver”.
- *verb-dir_trans-expletive_subj-lex*
  Exs.: “haver”.
- *verb-dir_trans-indef_null_obj-lex*
  Exs.: “chorar”, “negociar”, “matar”, “adotar”, “beber”.
- *verb-dir_trans-lex*
  Exs.: “odiar”, “elevar”, “freqüentar”, “exibir”, “alvejar”.
- *verb-dir_trans-opaque-lex*
  Exs.: “evitar”.
- *verb-dir_trans-subj_np_inf_cp+conj-lex*
  Exs.: “reflectir”, “refletir”, “desenvolver”, “afetar”, “afectar”.

- *verb-dir_trans_or_ind_trans-lex*
  Exs.: “presidir”.
- *verb-ditrans-lex*
  Exs.: “comprar”, “dar”, “entregar”, “oferecer”, “fornecer”.
- *verb-expletive_subj-comp_pp_de-lex*
  Exs.: “chegar”.
- *verb-expletive_subj-comp_que+vp-lex*
  Exs.: “haver”.
- *verb-identity_copula-lex*
  Exs.: “ser”.
- *verb-ind_trans-lex*
  Exs.: “mentir”, “obedecer”, “agradar”, “resistir”, “ladrar”.
- *verb-ind_trans_obligatory_comp-lex*
  Exs.: “dar”.
- *verb-individual_lvl_copula-lex*
  Exs.: “ser”.
- *verb-inherent_clitic-comp_pp_a-lex*
  Exs.: “abraçar”.
- *verb-inherent_clitic-comp_pp_a_obligatory-lex*
  Exs.: “fazer”.
- *verb-inherent_clitic-comp_pp_com-lex*
  Exs.: “abotoar”, “dar”.
- *verb-inherent_clitic-comp_pp_contra-lex*
  Exs.: “virar”.
- *verb-inherent_clitic-comp_pp_de-lex*
  Exs.: “deixar”.
- *verb-inherent_clitic-comp_pp_em-lex*
  Exs.: “perder”.
- *verb-inherent_clitic-expletive_subj-comp_pp_de-lex*
  Exs.: “tratar”.
- *verb-inherent_clitic-lex*
  Exs.: “suicidar”.
- *verb-intrans-comp_advp-lex*
  Exs.: “cheirar”.
- *verb-intrans-comp_pp_acercade_de_em_sobre-lex*
  Exs.: “falar”.
- *verb-intrans-comp_pp_acercade_de_sobre-lex*
  Exs.: “conversar”.

- *verb-intrans-comp_pp_com-lex*
  Exs.: “concordar”, “refilar”.
- *verb-intrans-comp_pp_com_de-lex*
  Exs.: “gozar”.
- *verb-intrans-comp_pp_com_obl-lex*
  Exs.: “dar”.
- *verb-intrans-comp_pp_de-lex*
  Exs.: “gostar”, “desaparecer”, “abusar”, “duvidar”, “precisar”.
- *verb-intrans-comp_pp_de_obligatory-lex*
  Exs.: “saber”.
- *verb-intrans-comp_pp_em-lex*
  Exs.: “bater”, “insistir”, “votar”.
- *verb-intrans-comp_pp_por-lex*
  Exs.: “passar”.
- *verb-intrans-comp_pp_sobre-lex*
  Exs.: “actuar”, “atuar”.
- *verb-intrans-lex*
  Exs.: “funcionar”, “fluir”, “voar”, “falir”, “correr”.
- *verb-intrans-obl_direction_em-lex*
  Exs.: “entrar”.
- *verb-intrans-obl_location-lex*
  Exs.: “morar”, “viver”.
- *verb-intrans-obl_origin-obl_direction-lex*
  Exs.: “ir”.
- *verb-obj_control-comp_pp_a-lex*
  Exs.: “convencer”, “obrigar”.
- *verb-obj_raising-comp_pp_a-lex*
  Exs.: “levar”.
- *verb-obj_raising-comp_pp_de-lex*
  Exs.: “impedir”.
- *verb-obj_raising-comp_vp-lex*
  Exs.: “fazer”, “ver”.
- *verb-obj_raising-comp_vp_or_cp+conj-lex*
  Exs.: “deixar”.
- *verb-optional_inherent_clitic-comp_pp_de-lex*
  Exs.: “rir”.
- *verb-predicative-comp_como-lex*
  Exs.: “ver”.

- *verb-predicative-cp_comp_alternation-lex*
  Exs.: “achar”.

- *verb-predicative-stage_level-cp_comp_alternation-lex*
  Exs.: “notar”, “ver”.

- *verb-predicative-stage_level-cp_or_vp_comp_alternation-lex*
  Exs.: “ouvir”.

- *verb-predicative-stage_level-lex*
  Exs.: “apanhar”, “encontrar”.

- *verb-predicative-stage_level-object_control-lex*
  Exs.: “deixar”, “pôr”.

- *verb-stage_lvl_copula-lex*
  Exs.: “estar”, “tar”.

- *verb-subj-comp_pp_a-obligatory_indir_obj-lex*
  Exs.: “saber”.

- *verb-subj-obligatory_comp_pp_a-obligatory_indir_obj-lex*
  Exs.: “ir”.

- *verb-subj_control-comp_np-comp_pp_a-lex*
  Exs.: “demorar”, “passar”.

- *verb-subj_control-comp_np_vp_or_cp_indicative-lex*
  Exs.: “saber”.

- *verb-subj_control-lex*
  Exs.: “lograr”, “tentar”, “querer”, “conseguir”, “aguentar”.

- *verb-subj_control-no_cp_comp-lex*
  Exs.: “experimentar”.

- *verb-subj_control-subj_np_inf_cp+conj-comp_vp_np_cp+ind-lex*
  Exs.: “salientar”.

- *verb-subj_control-vp_comp-lex*
  Exs.: “vir”.

- *verb-subj_np_inf-comp_pp_por+inf-lex*
  Exs.: “passar”.

- *verb-subj_np_inf_cp+conj-comp_indir_obj-lex*
  Exs.: “chegar”.

- *verb-subj_np_inf_cp+conj-comp_np_inf_cp_declarative-lex*
  Exs.: “significar”.

- *verb-subj_np_inf_cp+conj-inherent_clitic-comp_indir_obj_pp_a+inf_pp_a+cp-lex*
  Exs.: “dever”.

- *verb-subj_np_inf_cp+conj-intrans-comp_pp_com-lex*
  Exs.: “acabar”.

- *verb-subj_np_inf_cp+conj-trans-comp_pp_a*dat_de-lex*
  Exs.: “dispensar”.

- *verb-subj_np_inf_cp+conj-trans-comp_pp_em-lex*
  Exs.: “empatar”.

- *verb-subj_raising-comp_pp_a-aspectual_op_inchoative-lex*
  Exs.: “começar”, “desatar”, “passar”, “voltar”.

- *verb-subj_raising-comp_pp_a-lex*
  Exs.: “vir”.

- *verb-subj_raising-comp_pp_de-aspectual_op_process_to_culmination-lex*
  Exs.: “parar”.

- *verb-subj_raising-comp_pp_de-lex*
  Exs.: “acabar”, “haver”, “vir”.

- *verb-subj_raising-comp_pp_de_que-lex*
  Exs.: “ter”.

- *verb-subj_raising-comp_pp_por-lex*
  Exs.: “acabar”.

- *verb-subj_raising-comp_vp-aspectual_op_habitual-lex*
  Exs.: “costumar”.

- *verb-subj_raising-comp_vp-lex*
  Exs.: “dever”, “ir”, “poder”.

- *verb-subj_raising-comp_vp_or_cp-indir_obj-lex*
  Exs.: “parecer”.

- *verb-trans-comp_pp_a*dat_de-lex*
  Exs.: “extrair”.

- *verb-trans-comp_pp_a_em-lex*
  Exs.: “ressarcir”.

- *verb-trans-comp_pp_com-lex*
  Exs.: “coordenar”.

- *verb-trans-comp_pp_com_de-lex*
  Exs.: “fornecer”.

- *verb-trans-comp_pp_de-lex*
  Exs.: “cortar”, “receber”, “salvar”.

- *verb-trans-comp_pp_de-obligatory_comps-lex*
  Exs.: “ter”.

- *verb-trans-comp_pp_de_em-lex*
  Exs.: “pintar”.

- *verb-trans-comp_pp_em-lex*
  Exs.: “avaliar”, “injectar”, “injetar”.

- *verb-trans-comp_pp_por-lex*
  Exs.: “passar”.

- *verb-trans-obl_direction-lex*
  Exs.: “mandar”.

- *verb-trans-obl_direction_em-lex*
  Exs.: “enviar”, “lançar”.

- *verb-trans-obl_location-lex*
  Exs.: “anotar”, “armazenar”, “colocar”, “enterrar”, “meter”.

- *verb-trans-obl_origin-obl_direction-lex*
  Exs.: “passar”.

- *verb-unaccusative-ind_obj-lex*
  Exs.: “sobreviver”.

- *verb-unaccusative-inherent_clitic-comp_pp_com-lex*
  Exs.: “passar”.

- *verb-unaccusative-inherent_clitic-lex*
  Exs.: “constipar”, “engripar”.

- *verb-unaccusative-lex*
  Exs.: “crescer”, “desmaiar”, “cair”, “decorrer”, “adoecer”.

- *verb-unaccusative-obl_origin-obl_direction-lex*
  Exs.: “sair”, “chegar”, “fugir”, “vir”.

- *verb-unaccusative-optional_inherent_clitic-lex*
  Exs.: “passar”.

References

Adolphs, P., Oepen, S., Callmeier, U., Crysmann, B., Flickinger, D., and Kiefer, B. 2008. Some fine points of hybrid natural language parsing. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, Genoa, Italy.

Allegranza, V. 1998a. Determination and quantification. In Van Eynde, F. and Schmidt, P., editors, *Linguistic Specifications for Typed Feature Structure Formalisms*, pages 281–314. Office for Official Publications of the European Communities, Luxembourg.

Allegranza, V. 1998b. Determiners as functors: Np structure in italian. In Balari, S. and Dini, L., editors, *Romance in Head-driven Phrase Structure Grammar*, volume 75 of *CSLI Lecture Notes*, pages 55–108. CSLI Publications, Stanford.

Bach, E. 1986. The algebra of events. *Linguistics and Philosophy*, 9:5–16.

Baldwin, T., Beavers, J., Bender, E. M., Flickinger, D., Kim, A., and Oepen, S. 2005. Beauty and the beast: What running a broad-coverage precision grammar over the bnc taught us about the grammar—and the corpus. In Kepser, S. and Reis, M., editors, *Linguistic Evidence: Empirical, Theoretical and Computational Perspectives*, pages 49–70, Berlin. Mouton de Gruyter.

Bender, E. M., Flickinger, D., and Oepen, S. 2002. The Grammar Matrix: An open-source starter-kit for the development of cross-linguistically consistent broad-coverage precision grammars. In Carroll, J., Oostdijk, N., and Sutcliffe, R., editors, *Procedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei, Taiwan.

Binnick, R. I. 1991. *Time and the Verb: A Guide to Tense and Aspect.* Oxford University Press, Oxford.

Bonami, O. 2002. A syntax-semantics interface for tense and aspect in French. In Eynde, F. V., Hellan, L., and Beermann, D., editors, *The Proceedings of the 8th International Conference on Head-Driven Phrase Structure Grammar*, pages 31–50, Stanford. CSLI Publications.

Branco, A. and Costa, F. 2007. Identification and handling of dialectal variation with a single grammar. In Dirix, P., Schuurman, I., Vandeghinste, V., and Eynde, F. V., editors, *Proceedings of the 17th Meeting of Computational Linguistics in the Netherlands (CLIN17)*, pages 5–19, Utrecht. LOT.

Branco, A. and Henriques, T. 2003. Aspects of verbal inflection and lemmatization: Generalizations and algorithms. In Mendes, A. and Freitas, T., editors, *Proceedings of XVIII Annual Meeting of the Portuguese Association of Linguistics (APL)*, pages 201–210, Lisbon. APL.

Branco, A. and Silva, J. 2002. EtiFac: A facilitating tool for manual tagging. In *Proceedings of XVIII Annual Meeting of the Portuguese Association of Linguistics (APL)*, pages 81–90, Lisbon. APL.

Branco, A. and Silva, J. 2003a. Contractions: breaking the tokenization-tagging circularity. *Lecture Notes in Artificial Intelligence*, 2721:167–170.

Branco, A. and Silva, J. 2003b. A metric for the efficency of accurate tagging procedures. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2003)*, pages 71–77, Borovets, Bulgaria. Bulgarian Academy of Sciences.

Branco, A. and Silva, J. 2003c. Morpho-syntactic tagging without training corpus or lexicon: How far is it possible to get? In Mendes, A. and Freitas, T., editors, *Proceedings of XVIII Annual Meeting of the Portuguese Association of Linguistics (APL)*, pages 211–222, Lisbon. APL.

Branco, A. and Silva, J. 2003d. Portuguese-specific issues in the rapid development of state of the art taggers. In Branco, A., Mendes, A., and Ribeiro, R., editors, *Tagging and Shallow Processing of Portuguese: Workshop Notes of TASHA'2003*, pages 7–10, Lisbon. University of Lisbon, Faculty of Sciences, Department of Informatics.

Branco, A. and Silva, J. 2003e. Tokenization of Portuguese: resolving the hard cases. Technical Report TR-2003-4, Department of Informatics, University of Lisbon.

Branco, A. and Silva, J. 2004a. Evaluating solutions for the rapid development of state-of-the-art POS taggers for Portuguese. In Lino, M. T., Xavier, M. F., Ferreira, F., Costa, R., and Silva, R., editors, *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC2004)*, pages 507–510, Paris. ELRA.

Branco, A. and Silva, J. 2004b. Swift development of state of the art taggers for Portuguese. In Branco, A., Mendes, A., and Ribeiro, R., editors, *Language Technology for Portuguese: Shallow Processing Tools and Resources*, pages 29–45, Lisbon. Edições Colibri.

Callmeier, U. 2000. PET—A platform for experimentation with efficient HPSG processing techniques. *Natural Language Engineering*, 6(1):99–108. (Special Issue on Efficient Processing with HPSG).

Comrie, B. 1976. *Aspect: An Introduction to the Study of Verbal Aspect and Related Problems.* Cambridge University Press, Cambridge.

Comrie, B. 1986. Tense in indirect speech. *Folia Linguistica*, 20:265–296.

Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. 2001. Minimal Recursion Semantics: An introduction. *Language and Computation*, 3:1–47.

Copestake, A., Flickinger, D., Sag, I. A., and Pollard, C. 2005. Minimal Recursion Semantics: An introduction. *Journal of Research on Language and Computation*, 3(2–3):281–332.

Copestake, A. 2002. *Implementing Typed Feature Structure Grammars.* CSLI Publications, Stanford.

Costa, F. and Branco, A. 2012. Backshift and tense decomposition. In Müller, S., editor, *Proceedings of the 19th International Conference on Head-Driven Phrase Structure Grammar, Chungnam National University Daejeon*, pages 86–106.

Costa, F. 2013. *Processing Temporal Information in Unstructured Documents.* Ph.D. thesis, Universidade de Lisboa, Lisbon.

Davidson, D. 1967. The logical form of action sentences. In Rescher, N., editor, *The Logic of Decision and Action.* University of Pittsburgh Press.

de Swart, H. 1998a. Aspect shift and coercion. *Natural Language and Linguistic Theory*, 16:347–385.

de Swart, H. 1998b. *Introduction to Natural Language Semantics.* CSLI Publications, Stanford.

de Swart, H. 2000. Tense, aspect and coercion in a cross-linguistic perspective. In Butt, M. and King, T. H., editors, *Proceedings of the Berkeley Formal Grammar conference*, Stanford. CSLI Publications.

Dowty, D. R. 1979. *Word Meaning and Montague Grammar: the Semantics of Verbs and Times in Generative Semantics and Montague's PTQ.* Reidel, Dordrecht.

Ferreira, E., Balsa, J., and Branco, A. 2007. Combining rule-based and statistical methods for named entity recognition in portuguese. In *Actas do V Workshop em Tecnologia da Informação e da Linguagem Humana TIL.*

Flickinger, D. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6(1):15–28. (Special Issue on Efficient Processing with HPSG).

Flouraki, M. 2006. Constraining aspectual composition. In Müller, S., editor, *The Proceedings of the 13th International Conference on Head-Driven Phrase Structure Grammar*, pages 140–157, Stanford. CSLI Publications.

Garey, H. B. 1957. Verbal aspect in French. *Language*, 33(2):91–110.

Ginzburg, J. and Sag, I. A. 2000. *Interrogative Investigations: the Form, Meaning, and Use of English Interrogatives.* CSLI Publications, Stanford, California.

Ionin, T. and Matushansky, O. 2006. The composition of complex cardinals. *Journal of Semantics*, 23:315–360.

Kamp, H. and Reyle, U. 1993. *From Discourse to Logic: An Introduction to Modeltheoretic Semantics, Formal Logic and Discourse Representation Theory.* Kluwer, Dordrecht.

Kasper, R. T. 1996. The semantics of recursive modification. Manuscript.

Krifka, M. 1992. Thematic relations as links between nominal reference and temporal constitution. In Sag, I. and Szabolcsi, A., editors, *Lexical Matters*, pages 29–53. CSLI Publications, Chicago University Press.

Web Page. LXConjugator. http://lxconjugator.di.fc.ul.pt.

Web Page. LXLemmatizer. http://lxlemmatizer.di.fc.ul.pt.

Web Page. LXSuite. http://lxsuite.di.fc.ul.pt.

Manning, C. 1992. Presents embedded under pasts. Manuscript.

Michaelis, L. 2006. Tense in English. In Aarts, B. and McMahon, A., editors, *The Handbook of English Linguistics.* Blackwell, Oxford.

Michaelis, L. A. 2011. Stative by construction. *Linguistics*, 49:1359–1400.

Moens, M. and Steedman, M. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15–28.

Moens, M. 1987. *Tense, Aspect and Temporal Reference.* Ph.D. thesis, Center for Cognitive Science, University of Edinburgh.

Müller, A. 2002. The semantics of generic quantification in Brazilian Portuguese. *Probus: International Journal of Latin and Romance Linguistics*, 14(2):279–298.

Munn, A. and Schmitt, C. 1998. Against the nominal mapping parameter: Bare nouns in Brazilian Portuguese. In *Proceedings of NELS 29*, pages 339–353, Amherst, MA. GLSA, The University of Massachusetts.

Nerbonne, J. and Mullen, T. 2000. Null-headed nominals in German and English. In van Eynde, F., Schuurman, I., and Schelkens, N., editors, *Proc. of Computational Linguistics in the Netherlands 1998*, pages 143–64.

Oepen, S., Netter, K., and Klein, J. 1997. TSNLP—Test suites for natural language processing. *Linguistic Databases, CSLI Lecture Notes*, 77:3–28.

Oepen, S. 2001. [incr tsdb()]—competence and performance laboratory. User manual. Technical report, Computational Linguistics, Saarland University, Saarbrücken, Germany.

Parsons, T. 1990. *Events in the Semantics of English: A Study in Subatomic Semantics.* Number 19 in Current Studies in Linguistics Series. MIT Press, Cambridge, MA.

Partee, B. 1973. Some structural analogies between tenses and pronouns in English. *The Journal of Philosophy*, 70:601–609.

Platzack, C. 1979. *The Semantic Interpretation of Aspect and Aktionsarten. A Study of Internal Time Reference in Swedish.* Foris, Dordrecht.

Pollard, C. and Sag, I. 1987. *Information-Based Syntax and Semantics, Vol. 1.* CSLI Publications, Stanford.

Pollard, C. and Sag, I. 1994. *Head-Driven Phrase Structure Grammar.* Chicago University Press and CSLI Publications, Stanford.

Pullum, G. K. 1975. *People* deletion in English. In *Working Papers in Linguistics*, volume 14, pages 95–101. Ohio State University.

Pustejovsky, J., Littman, J., Saurí, R., and Verhagen, M. 2006. Timebank 1.2 documentation. `http://timeml.org/site/timebank/documentation-1.2.html`.

Pustejovsky, J. 1991. The syntax of event structure. *Cognition*, 41:47–81.

Reichenbach, H. 1947. *Elements of Symbolic Logic.* University of California Press, Berkeley.

Rodríguez, J. P. 2004. *Interpreting the Spanish* Imperfecto*: Issues of Aspect, Modality, Tense, and Sequence of Tense.* Ph.D. thesis, The Ohio State University, Columbus, Ohio.

Ross, J. R. 1967. *Constraints on variables in syntax.* Ph.D. thesis, Massachusetts Institute of Technology.

Sag, I. A., Wasow, T., and Bender, E. M. 2003. *Syntactic Theory—A Formal Introduction.* CSLI Publications, Stanford.

Schlenker, P. 2004. Sequence phenomena and double access readings generalized (two remarks on tense, person and mood). In Lecarme, J. and Guéron, J., editors, *The Syntax of Time.* MIT Press, Cambridge, Mass.

Smith, C. S. 1997. *The Parameter of Aspect.* Kluwer Academic Publishers, Dordrecht.

Van Eynde, F. 1998. Tense, aspect and negation. In Van Eynde, F. and Schmidt, P., editors, *Linguistic Specifications for Typed Feature Structure Formalisms. Studies in machine Translation and Natural language Processing*, volume 10, pages 209–280. Luxembourg.

Van Eynde, F. 2000. A constraint-based semantics for tenses and temporal auxiliaries. In Cann, R., Grover, C., and Miller, P., editors, *Grammatical interfaces in HPSG*, pages 231–249, Stanford University. CSLI Publications.

Van Eynde, F. 2003a. On the notion 'determiner'. In Müller, S., editor, *Proceedings of the HPSG-2003 Conference, Michigan State University, East Lansing*, pages 391–396, Stanford. CSLI Publications.

Van Eynde, F. 2003b. Prenominals in Dutch. In Kim, J.-B. and Wechsler, S., editors, *The Proceedings of the 9th International Conference on HPSG*, Stanford University. CSLI Publications.

Vendler, Z. 1957. Verbs and times. *The Philosophical Review*, 66:143–160.

Vendler, Z. 1967. Verbs and times. In *Linguistics in Philosophy*, pages 97–121. Cornell University Press, Ithaca, New York.

Verkuyl, H. J. 1972. *On the Compositional Nature of the Aspects.* Reidel, Dordrecht.

Verkuyl, H. J. 1993. *A Theory of Aspectuality: The interaction between temporal and atemporal structure.* Cambridge University Press, Cambridge.