AUTOMATIC PARSING OF PORTUGUESE

Eckhard Bick

Department of Linguistics, Århus University, Nordre Ringgade, DK-8000 Århus C

tel: +45 - 89 422170, fax: +45 - 86 281397, e-mail: lineb@hum.aau.dk

Abstract

The paper presents an automatic parser for unrestricted Portuguese text, ultimately intended for applications like corpora tagging, grammar teaching and machine translation. The parser uses a hand built high coverage lexicon to assign morphological tags (base form, word class and inflection alternatives) to every wordform in the text, and then disambiguates multiple readings (on average, 2.08) by using grammatical rules formulated in the Constraint Grammar formalism. On the next level of analysis, tags for syntactical form and function alternatives are mapped onto the wordforms and disambiguated in a similar way. In spite of using a highly differentiated tag set, the parser yields correctness rates - on running unrestricted and unknown text - of over 99% for morphology/POS and 97-98% for syntax (where work is still in progress), even when geared to full disambiguation. Among other things, argument structure, dependency relations and subclause function are treated in an innovative way. The parser uses valency and semantical class information from the lexicon, but disambiguation on these levels is still experimental.

The system runs at about 100 words/sec on a 100 MHz Pentium based Linux system, when using all levels. Morphological and POS disambiguation alone approach 1000 words/sec.

1 Overview

In this paper I am going to present a running parser for unrestricted Portuguese text, which has been developed over serveral years in the context of my Ph.D. research on the Computational Analysis of Portuguese, at Århus University, Denmark. The project has a lexicographic base (treated in my MA Thesis) and a Machine Translation perspective, but here I want to focus on the parser as such - that is, the morphological and POS tagging and the syntactical parse. I shall try to define and exemplify the scope and descriptive power of the parser, as well as explain some of the underlying principles used for tagging and disambiguation. Though the project is not finished yet, an evaluation of the present performance, in terms of precision and recall, will be undertaken.

2 Background

Two particularly recalcitrant problems in NLP are (a) the well-known fact that "all grammars leak" (a fact haunting especially Prolog style DCG-grammars and unification grammars) and (b) the large amount of ambiguity resulting from any more detailed NLP-description (a problem common to most parsers, but especially obvious in probabilistic approaches to tree-structured syntactic analysis).

The *constraint grammar* formalism (CG), as developed by the Helsinki school (e.g. Karlsson et.al., 1995), addresses both problems. Implemented as a FSM-compiler, it applies a grammar of constraint rules to text which has been automatically tagged with all *possible* part-of-speech tags by a lexicon-based morphological analyser.

```
(1)        "<revista>"
              "revista"  <+n> <CP> <rr> N F S
              "revestir"  <vt> <de^vtp> <de^vrp>  V PR 1/3S SUBJ VFIN
              "revistar"  <vt>  V IMP 2S VFIN
              "revistar"  <vt>  V PR 3S IND VFIN
              "rever"  <vt> <vi>  V PCP F S
```

With a CG-term, such an ambiguous list of readings is called a *cohort*. In the example, the word form 'revista' has one noun-reading (female singular) and four (!) verb-readings, the latter covering three different base forms, subjuncitve, imperative, indicative present tense and participle readings. Conventionally, POS and morphological features are regarded as primary tags and coded by capital letters. In addition there can be secondary lexical information about valency and semantical class, marked by <> bracketing.

A *constraint grammar rule* brings the ambiguity problem to the foreground by specifying which reading (out of a cohort of ambiguous readings for a given word) is impossible (and thus to be discarded) or mandatory (and thus to be chosen) in a given sentence-context. For instance, a rule might discard a finite verb reading after a preposition (2a) , or when another - unambiguous - finite verb is already found in the same clause, with no coordinators present (2b).[1]

(2a) @w =0 (VFIN) (-1 PRP)

[= discard (=0) any reading (@w) that is (VFIN) if the first word to the left (-1) is a preposition PRP)]

(2b) @w =0 (VFIN) (*1C VFIN *L) (NOT *L CLB/KC) (NOT *-1 CLB-WORD)

[1] Ordinarily, this disambiguation process works on whole cohort lines, i.e. distinguishes between POS, base form and inflection, but tolerates competing valency options. However, on a higher level of analysis, I have introduced valency and semantical disambiguation, too. This can be very useful for polysemy resolution, like in "rever", where the transitive <vt> - intransitive <vi> distinction has a meaning correlate: 'tornar a ver' [see again] vs. 'transudar' [leak through]. Likewise, "revista" followed by a name <+n> or being read (semantical class <rr>) is more likely to be a newspaper than an inspection (semantical class <CP> for action: +CONTROL, +PERFECTIVE).

[= discard VFIN if there is another finite verb (VFIN) anywhere to the right (*1) and unambiguous (C) with no clause-boundary (CLB) og coordinating conjunction (KC) interfering to its relative left (*L). Discard only if there is no subordinator (CLB-WORD) anywhere to the left (*-1)]

By applying the rule set several times, more and more words in the sentence become unambiguous, and in the end, only one reading is left for every word. Since the individual rule can be made very "cautious" by adding more context contidions, and since the last surviving reading will never be discarded, the formalism is very robust. Even imperfect input will yield *some* parse. Unlike probabilistic systems, where "manual interference" as in the introduction of bias on behalf of irregular phenomena often has an adverse side-effect on the overall performance of the parser (due to interference with the ordinary statistical "rules" based on the *regular* "majority" phenomena), Constraint Grammar tolerates and even encourages the incremental "piecemeal " addition of exceptions and context conditions for individual rules (For a comparison of statistical and constraint-based methods see Chanod & Tapanainen, 1994).

Constraint grammars have been developed for English (e.g. Karlsson et. al, 1991) and several other languages (there is a commercial version available for English, and work in progress on different languages from both the germanic, romance and finno-ugric families, e.g. German, French, Swedish, Danish, Finnish and, of course, Portuguese) . Grammars typically consist of 1-2000 rules, consistently yielding correctness rates of over 99% for morphological and part-of-speech tagging.

By mapping possible syntactical functions on POS-tags from the morphological module, Constraint Grammar can also be used for syntactic parsing, as in the Bank of English project where 200 million words were analysed (Järvinen, 1994).

3 More descriptive power for Constraint Grammar syntax

In its essence CG is a robust disambiguating philosophy, which does not build a specific sentence structure, but carves away what cannot be part of any structure. That way neither the carving method (the rule system) nor the carving tools (rule compilers) are determined by the Constraint Grammar idea as such. And even less the finished sculpture. Every carpenter is free to apply his own beauty ideals. Or isn't he?

Historically, CG has its roots in morphological analysis, most systems run with a two-level morphological analyser (cp. Koskenniemi, 1983) as preprocessor, and focus on morphological features and parts of speech. Therefore, information is traditionally word-bound and coded as tags (to be attached to words). "Flat" grammar is a natural consequence of this. Without special dependency links such a flat syntactical description works fine only as long as individual words bear all of a phrase's functional burden. The description gets into trouble when higher level dependencies are involved. Thus, a CG-description without subclause-[function]-tags is bound to suffer from shortcomings like the following:

1. Clause boundary markers (or their rule context equivalents) are not hierarchically motivated, so there may be problems with unclear clause continuation after, e.g., center embedded relative clauses.

2. Certain valency features may be left "unsatisfied", e.g. missing subjects in English (*'Visiting the Louvre was not his only reason for coming to Paris')*, or missing accusative objects ('that/que'-clauses after "cognitive" verbs).

3. Surplus arguments due to unclear clause level resolution, like in '*..., o baque foi atenuado pelo fato de sua mulher ter um emprego que garante as despesas básicas da família',* where both *baque* and *mulher* are subjects, but the second subject can only be fully structuralized by attaching its main verb *(ter)* as clausal infinitive argument to the preceding preposition *(de).*

4. Reduced dependency information content as compared to tree structures.

I believe that, by distinguishing between CG as a disambiguation technique, on the one hand, and the descriptional system to be carved, on the other hand, some kind of flat representation can be designed that is functionally equivalent to tree structures, and can express argument and valency structures in a hierarchical way.

My approach, aimed at solving the above problems, has been to add attachment direction markers to all argument tags (<, >, pointing to the head in question, or, on clause level, to the main verb), and to apply double tags to the central linking word in subclauses , - that is, to the “complementizer” (subordinating conjunction, relative or interrogative) in finite and absolute subclauses, and to the infinitive, gerund or participle in infinite subclauses[2]. These words, then, bear both an “internal” tag (@...) which describes their function inside the subclause, and an “external” tag (@#...), that describes the function of the subclause as a whole when integrated into the next higher level in the sentence’s clause hierarchy. Technically, the disambiguation process works on two lists of @- and @#-tags, respectively, so that internal and external function tags can be treated individually:

(3)	Sabe	"saber" <vq> V PR 3S IND	@FMV	
	que	"que" KS	**@#FS-<ACC**	**@SUB**
	os	"o" <art> DET M P		@>N
	problemas	"problema" N M P		@SUBJ>
	são	"ser" <vK> V PR 3P IND		@FMV
	graves	"grave" ADJ M/F P		@<SC
	$.			

[where: @FMV = finite main verb, @#FS-<ACC = finite subclause, functioning as direct (accusative) object attached to a main verb to the left, @SUB = subordinator, @>N = prenominal modifier, @SUBJ> = subject for a main verb to the right, @<SC = subject complement for a (copula) verb to the left, V = verb, KS = subordinating conjunction, DET = determiner, N = noun, ADJ = adjective, PR = present tense, IND = indicative, 3S = third person singular, 3P = third person plural, M = male, F = female, S = singular, P = plural, <art> = article, <vq> = cognitive verb, <vK> = copula verb]

In this way both form (phrase constituents like @>N for prenominals and @N< for postnominals) and function (argument roles like @SUBJ>, @<ACC, @N<PRED for phrases or @#ICL-SUBJ>, @#FS-<ACC for clausal units) are specified, creating a kind of "flat tree structures" that can be shown to be nearly equivalent to traditional tree structures in terms of information content. Exceptions are the sometimes ambiguous attachment of postnominals, as well as some cases of coordination and free nominal adjuncts. But even this remaining degree of underspecification may be regarded as advantageous: postnominal PP's like '*from France*' in '*the man with the hat from France*', for example, will simply be described as attached to an NP to the left (*from* PRP @N< *France* PROP @P<). Since it is hard to see how *any* primarily syntactic description should totally resolve this kind of ambiguity, elegant (=flat) underspecification might be the best solution.

4 A teleological judgement perspective

When comparing different syntactic descriptions, information content and constituent structure are only two of all the possible judgement perspectives, and both are motivated by certain theoretical backgrounds, like functional or generative grammar. It may be more revealing, however, to take into account which uses a certain description most likely will be put to.

Here, my own perspective is machine translation, and thus, the following aspects become important:

* Detailed word order independent functional tags make it easier to transform source language structure into target language structure, without to many complicated transformation rules. Especially where languages like Portuguese are involved, that - unlike English - permit almost any order of clause level arguments.

* It is of great importance for polysemy resolution to know which of a word’s potential valency patterns has been instantiated in a given clause or phrase, and which semantic class fills a given valency slot. Therefore, valency tags (and selection restrictions) are motivated not only as secondary tags (used

[2] For another approach to subclause function tagging, proposed for English, see Voutilainen (1994), where the subclause's main verb is assigned an additional tag (...@) in a similar way. The dependency information gap, on the other hand, is here approached by assigning clause boundary tags and by distinguishing between arguments of finite and non-finite main verbs respectively.

to disambiguate syntactic alternatives), but also as primary tags, which can be subjected to disambiguation themselves.

* The above mentioned underspecification of postnominals, coordination and free nominal adjuncts becomes an asset when seen from a machine translation perspective: - first, a large part of these cases is “true ambiguity”, which can only be resolved by the fully contextualized listener/reader. In any case, it is “true *syntactic* ambiguity”. - Second, some of these structural ambiguities (prepositional phrase attachment and coordination) are fairly universal, i.e. language independent, so that they can be preserved in translation. Making such ambiguiy explicit would only put an unnecessary burden on the translation module. (Adjective attachment, either postnominally or as free adjuncts, is more problematic due to possible agreement links between head and modifier).

5 The Portuguese parser

5.1 The tag set

The parser´s tag set contains 13 part of speech categories that combine with 24 inflection tags, yielding hundreds of distinct complex tag strings. In the tag 'V PR 3S IND VFIN', for instance, 'V' (the word class) alternates with the 12 other word classes, and inside the V-class the 'PR' (present tense) alternates with 5 other tenses, which each can apper in 6 person-number forms of either IND (indicative) or SUBJ (subjunctive). Thus, there are 6 x 6 x 2 = 72 tense bearing finite verb forms expressed by only 6+6+2 =14 partial tags. This analytical nature of the tag string is a great advantage for readability as well as for writing disambiguation rules. Unlike some other systems (cp., for example, the CLAWS system, as in Leech, Gaside, Bryant, 1994), there is a clear distinction between base forms ("words"), POS-classes and inflection features. Also, POS-definitions are almost completely morphological, and kept apart from syntactically motivated categories. Thus, a noun (N) is defined paradigmatically as *that* word-class that has gender as a (fixed) lexeme category and number as a (variable) wordform category. The inverse is true for numerals (NUM), whereas in proper nouns (PROP) both number and gender are normally lexeme categories, and in adjectives, (ADJ) both are word form categories[3].

The syntactical tag set includes about 40 tags for word/phrase function and about 30 tags for clause function (covering three classes of clauses: finite subclauses, infinite subclauses and absolute [= verb-less] subclauses). Again, the real number of distinct tag strings is much higher, since the word bearing the clause function also has to be marked for its clause-internal function.

Due to the experimental nature of the valency and semantics subsystems, no fixed number of tags can be given at the time of writing. Approximate numbers are about 100 for valency classes (especially for verbs), and 200 for semantical classes (mostly for nouns). Semantical classes are built from 16 "atomic" features (like, for instance, ±HUM).

5.2 Levels of analysis and program modules

The parser as such, in its present version, consists of the following modules:

♦ 1. a **morphological analyser** (written in C, described in Bick, 1995), which treats part-of-speech, inflection, derivation, fixed expressions and incorporating verbs. The analyser uses a hand-made lexicon of 70.000 entries representing some 50.000 lexemes (adapted from the author's lexicographic MA thesis, Bick, 1993)

[3] Interestingly, pronouns can be divided by the same scheme, yielding determiners (DET) with the same (inflectable) categories as adjectives and "specifiers" (SPEC: indefinite pronouns, nominal quantifiers, nominal relatives), with the same (uninflectable) categories as proper nouns. Personal pronouns (PERS), finaly, have four word form categories: number, gender, case and person. All three pronominal classes are distinct from the "real" nominal by the morphological fact, that only the latter can be used for derivation. In this scheme, 'o' and 'este' are always just determiners, whether they are used pre-nominally or not. A tag for article (<art>) *is* provided for 'o' , but it is not a POS-category, and is only disambiguated at a later stage (the valency level).

Participles (V PCP), the enfant terrible of POS-categories, are morphologically marked as verb-derivatives ('-id/-ad'), but then, outside the verb chain, they have the same inflection categories as adjectives. Therefore, in these cases, the parser collapses any PCP/ADJ-ambiguity into one reading: <ADJ> V PCP.

- ♦ 2. a **morphological disambiguator** using 1700 Constraint Grammar rules
- ♦ 3. a **syntactical "mapper"** which assigns possible syntactical tags, using 400 context based function assignment rules
- ♦ 4. a **syntactical disambiguator** using 1500 Constraint Grammar rules
- ♦ 5. a **valency disambiguator** and **semantical class diambiguator** (2200 Constraint Grammer rules, not fully operational)

On top of the Portuguese parser a machine translation system is being constructed, featuring the following modules:

- ♦ 6. bilingually motivated **polysemy resolution**, based on disambiguated morphological, syntactical, valency and semantic class tags
- ♦ 7. a base form **translation module** Portuguese-Danish (written in C)
- ♦ 8. a **bilingual syntax transformation module** (written in Perl) rearranging source language (Portuguese) word order, phrase and clause structure according to target language grammar (Danish)
- ♦ 9. a Danish **morphological generator** (written in C) that works on - translated - base forms and tag lists and builds Danish words from a base form lexicon with inflection information

The context window for all modules is the sentence, that means grammar rules can "look" at the full clause hierarchy, but not past a full stop. The only exception to this is some experimental anaphora resolution in module 8.

5.3 Example parse

When running all parsing levels and the first two MT modules (i.e., up to level 7), a full analysis yields "verticalised" sentences like the one below. Of course, when what is asked for is a "mere" POS/morphological tagging, or a "pure" syntactical analysis, many of the <..> - bracketed tags included below would become irrelevant, and should be removed from the final output.

(4)	...	
	$,	
	o	[o] <art> DET M S **@>N** 'den'
	baque	[baque] <cP> N M S **@SUBJ>** 'fald'
	foi	[ser] <x+PCP> V PS 3S IND VFIN **@FAUX** 'blive'
	atenuado	[atenuar] <vt> <sN> V PCP M S **@IMV @#ICL-AUX<** 'svække'
	por	[por] <sam-> <+INF> <PCP+> PRP **@<PASS** 'af'
	o	[o] <-sam> <art> DET M S **@>N** 'den'
	fato	[fato] <ac> <+de+INF> N M S **@P<** 'kendsgerning'
	de	[de] PRP **@N<** 'af'
	sua	[seu] <poss 3S/P> DET F S **@>N** 'hans'
	mulher	[mulher] <H> N F S **@SUBJ>** 'kvinde'
	ter	[ter] <vt> <sH> V INF 0/1/3S **@IMV @#ICL-P<** 'have'
	um	[um] <quant2> <arti> DET M S **@>N** 'en'
	emprego	[emprego] <stil> <ac> N M S **@<ACC** 'stilling'
	que	[que] <rel> SPEC M/F S/P **@SUBJ> @#FS-N<** 'som'
	garante	[garantir] <vt> <v-cog> V PR 3S IND VFIN **@FMV** 'garantere'
	as	[a] <art> DET F P **@>N** 'den'
	despesas	[despesa] <ac> N F P **@<ACC** 'udgift'
	básicas	[básico] <jn> ADJ F P **@N<** 'basal'
	de	[de] <sam-> PRP **@N<** '(genitiv)'
	a	[a] <-sam> <art> DET F S **@>N** 'den'
	família	[família] <HH> N F S **@P<** 'familie'
	$.	

In the notation used, each wordform is followed by its baseform [...], valency and semantical information <...>, part of speech and inflection (in CAPITAL LETTERS), syntactical form and function (**@** for words and phrase heads, **@#** for clausal units), and finally the chosen base form translation '...'.

[**POS:** DET = determiner, N = noun, V = verb, PRP = preposition, ADJ = adjective, SPEC = specifier (uniflected "substantival" pronoun), **inflection:** M = male, F = female, S = singular, P = plural, VFIN = finite verb, IND = indicative, PCP = participle, PR = present tense, 3S = third person singular, **syntactical word class:** <art> = (definite) article, <arti> = indefinite article, <poss> = possessive, <quant2> = quantifier, <rel> = relative, **orthography:** <sam-> <-sam> = word split by the parser, here: 'pelo' and 'da', **syntax:** @>N = prenominal, @SUBJ> = subject, @FAUX = finite auxiliary, @IMV = infinite main verb, @#ICL-AUX< = infinite subclause as argument of auxiliary, @<PASS = passive agent, @P< = argument of preposition, @#ICL-P< = infinite subclause as argument of preposition, @<ACC = accusative object, @#FS-N< = finite subclause as postnominal (relative clause), @FMV = finite main verb, **valency:** <x+PCP> auxiliary with participle valency, <vt> transitive verb, **semantical class:** <v-cog> cognitive verb, <cP> = event (-CONTR, +PERF), <ac> = abstract countable, <stil> = job position, <HH> human group term, **selection restrictions:** <jn> = non-human adjective, <sN> = combines with non-human subject, <sH> combines with human subject)]

5.4 Technical performance and corpus base

The parser runs at about 100 words/sec on a 100 MHz Pentium based Linux system, when using all levels. Morphological and POS-disambiguation alone approach 1000 words/sec.

For training the parser I have built a hand-tagged bench mark corpus of 33.000 words, as well as used parts of the (untagged) mixed Borba-Ramsey Corpus (670.000 words) and news articles from VEJA (600.000 words). The primary language variety is Brazilian Portuguese, but also European Portuguese can be handled.

Test texts from the research community (in ISO Latin-1 or Macintosh format) can be automatically analysed by e-mail at *eckhard@ling.hum.aau.dk*. For details, contact the author.

6 Evaluation

6.1 Morphological and POS analyser

Allowing for a heuristical solution for unknown proper nouns, the lowest level of analysis, the morphological analyser, has a succes rate of 99.6 - 99.7 % (on the mixed Borba-Ramsey Corpus), i.e. 0.3 % of non-name word forms in running text do not have their base form registered in the lexicon. In a smaller corpus with the same error rate (132.000 words of literature excerpts and secondary prose literature from the RNP depository of Brazilian literature), the following error distribution was found:

non-portuguese words and texts passages (especially English)		38.4 %
non-capitalised names and abbreviations (e.g pharmaceutical names)		6.1 %
more or less automatically correctable ortographic variation and typing errors		33.0 %
base form not found		19.7 %
a) base form listed in Aurelio	15.1 %	
b) base form not listed in Aurelio	4.6 %	
derivation/flexion analysis failure		2.5 %
other		0.3%

The figures indicate that in terms of "unanalysable Portuguese non-name words", lexicon and analysis failures can come down to 0.1 % of running word forms. Especially corpus mark-up for foreign quotes, English loan words and scientific latinid terms would improve performance. In my parser, a morphological "guesser" compensates for the lexicon failures, using flexion information, derivative affixes and so on. In this way, the tag cohort that is passed on to the CG-disambiguatior, does nearly always contain the correct reading. More than half the word forms, though, get more than one reading (on average, 2.08), and this is where the CG-disambiguator , the next level of analysis, comes into the picture.

6.2 Disambiguation

6.2.1 Training texts

Working on "known" bench mark texts of 10-20.000 words, by constantly testing rule performance on manually introduced <Correct!> - markers, the Portuguese morphological tagger (analyser and disambiguator together) can be geared to resolve nearly all ambiguity while retaining a 99.9 % corrrectness rate. For unknown texts the results are obviously lower, yet, the result is not irrelevant, since it shows that the CG approach does not suffer from system immanent interference problems to the same degree as, say, a probabilistic tagger based on a pure trigram HMM, where (to my knowledge) even retraining and measuring on the same corpus seldom yields more than 97% correctness, even for parts of speech.

Aiming at maximal precision, I have also worked on a larger, untagged text (170.000 word from the Borba-Ramsey corpus) on both the morphological and syntactical levels. This was an option, since *precision* (defined as the percentage of surviving readings, that are correct) can be approximated by minimising ambiguity, at least as long as intermittent bench mark runs ensure that new rules discard few correct readings, and ambiguity is still fairly high. Surviving ambiguity, then, can easily be measured without manual control on any text corpus. In contrast, *recall* (defined as the percentage of correct readings, that survived the disambiguation) has - in the absence of a large tagged Portuguese corpus for measuring - to be calculated on smaller sample texts. Here, when forcing the parser into full disambiguation, where all words - with the exception of the rare cases of true ambiguity - end up with one reading only, one can regard the recall figure as a direct measure for the parser's performance, and I will henceforth use the more general term *correctness* to mean *recall at 100% disambiguation.*

6.2.2 Test texts

Though my project is not finished yet, I have done some such correctness evaluation on unknown texts, too. These test runs, while being fairly small, consistently suggest a correctness rate of over 99% for morphology and part of speech, when analysing unknown unrestricted text. For syntax the figures are 98% for classical literary prose (Eça de Queiroz, "O tesouro") and 97% for the more inventive journalese of newspaper texts (VEJA, 9.12.1992), as shown in the table below.

One might assume that errors are evenly spread throughout the text, which would - for an average sentence length of 15 words - mean about one morphological error in every tenth sentence and a syntactical error in every third. However, this is not true: errors appear in clusters, obviously most morphological errors also appear in the list of syntactical errors, and many syntactical errors interfere with readings in their neighbourhood, due to rules that depend on clause boundary words, uniqueness principle and so forth. Thus, a V-N word class error can cause cause 2 or 3 syntactical errors around it. This clustering tendency of syntactical errors is good news both the overall robustness of the result (there are many sentences, that are completely error free), and for the work of the grammarian: mending the grammar at one point may remove a whole chain of secondary interference errors. Likewise, when seen in isolation, - that is, when supplied with error-free morphological input -, the syntactical parser on its own can yield even better results. Thus, for VEJA newspaper texts, the correctness rate will rise by 0.5-1 %.

	O tesouro		*VEJA 1*		*VEJA 2*	
	ca. 2500 words		ca. 4800 words		ca. 3140 words	
	errors	correct-ness	errors	correct-ness	errors	correct-ness
Part-of-speech errors	16		15		24	
Base-form & flexion errors	1		2		2	
All morphological errors	17	**99.3 %**	17	**99.7 %**	26	**99.2 %**
syntactical: word & phrases	54		118		101	
syntactical: subclauses	10		11		13	
All syntactical errors	64	**97.4 %**	129	**97.3 %**	114	**96.4 %**
"local" syntactical errors due to POS/morphological errors	- 27		- 23		- 28	
Purely syntactical errors	37	**98.5 %**	106	**97.8 %**	86	**97.3 %**

6.3 Text type interference and tag set complexity

Obviously, evaluation figures like these will be heavily text type and corpus mark-up dependent. In my VEJA texts, for example, special features like the following can be found:

* lots of headlines without finite verbs, and with upper case letters only
* unclear sentence boundaries due to headlines without punctuation marks, quotes, referred speech, and all kinds of bracketing for parenthetic information
* many unknown personal and place names, often abbreviated
* English expressions like "bad boy", "joystick" and "zumbi"
* "journalese" constructions with multi-layered sentences, including lots of bound and free predicatives, appositions and so on, all acting as false "argument candidates" in the clause structure

However, none of the above problems are in principle intractable for the CG-approach, and by providing for special features like these in my rule set (and lexicon) I hope to be able to reduce the error rate substantially by the end of my project.

Also, when comparing the above correctness figures to the results of other approaches, one has to bear in mind the complexity of the tag set and the information content of the categories used. Thus the attachment and functional information, that my parser provides for prepositional phrases (such as post-nominal adject @N< , post-adjectival/adverbial adject @A<, adjunct adverbial @<ADVL, @ADVL>, @ADVL, adverbial object @<ADV, @ADV>, prepositional object @<PIV, @PIV>, subject complement @<SC, free predicative @<PRED, complementiser argument @AS<) can potentially give rise to numerous errors, that would just not be visible if all these tags were collapsed into a bare syntagmatical 'PP' (prepositional phrase) or a rudimentary "functional" 'ADVL' (adverbial).

7 Outlook

Constraint Grammar based parsers are fast, robust and yield descriptively elegant output. Portuguese, a highly inflecting language with a relatively free word order, seems to involve the same degree of rule complexity as found for English, a fixed word order language with little inflection, corroborating the universality claim of the CG approach. With similar parsers having been successfully integrated into applications like spelling checkers and research oriented tagged text banks on a morphological/POS level, as a next step, Machine Translation seems to be a promising field, if the approach - as my research indicates - can be shown to be able to handle syntactic tree structures, valency and polysemy resolution as well.

References

Eckhard Bick, *Portugisisk - Dansk Ordbog,* Mnemo, Århus, 1993, 1995

Eckhard Bick, *The Parsing System "Palavras", Documentation,* unpublished Ph.D. project evaluation, 1995, updated version forthcoming

Jean-Pierre Chanod & Pasi Tapanainen, "Tagging French - comparing a statistical and a constraint-based method", adapted from:*Statistical and Constraint-based Taggers for French,* Technical report MLTT-016, Rank Xerox Research Centre, Grenoble, 1994

Timo Järvinen, "Annotating 200 million words: The Bank of English project", in *Proceedings of The 15th International Conference on Computational Linguistics Coling-94,* Kyoto, Japan, 1994 (cited from: Pasi Tapanainen, *The Constraint Grammar Parser CG-2*, Publications No. 27, Department of Linguistics, University of Helsinki, 1996)

Fred Karlsson, Atro Voutilainen, Juka Heikkilä, Arto Anttila (eds.), "Constraint Grammar, A Language-Independent System for Parsing Unrestricted Text, with an application to English", in: *Natural language text retrieval. Workshop notes from the Ninth National Conference on Artificial Intelligence,* Anaheim, CA, American Association for Artificial Intelligence, 1991

Fred Karlsson, Atro Voutilainen, Juka Heikkilä, Arto Anttila (eds.), *Constraint Grammar, A Language-Independent System for Parsing Unrestricted Text,* Mouton de Gruyter, Berlin 1995

Fred Karlsson, "Robust parsing of unconstrained text", pp. 97-121, in: Nellike Oostdijk & Pieter de Haan, *Corpus-based research into language,* Amsterdam, 1994

Kimmo Koskenniemi, *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production,* Publication No. 11, Department of Linguistics, University of Helsinki, 1983

Geoffrey Leech, Roger Garside, Michael Bryant, "The large-scale grammatical tagging of text", pp. 47-64, in: Nellike Oostdijk & Pieter de Haan, *Corpus-based research into language,* Amsterdam, 1994

Atro Voutilainen, Juka Heikkilä, Arto Anttila, *Constraint Grammar of English, A Performance-Oriented Introduction,* Publication No. 21, Department of General Linguistics, University of Helsinki, 1992

Atro Voutilainen, *Designing a Parsing Grammar,* Publications No. 22, Department of Linguistics, University of Helsinki, 1994