Reusing the CG-2 Grammar for Processing Basque Complex Postpositions

Reutilización de una Gramática de Restricciones (CG-2) para el procesamiento de las posposiciones complejas en Euskera

José María Arriola, Itziar Aduriz, Izaskun Aldezabal, María Jesús Aranzabe, Clara Ceberio, Ainara Estarrona, Mikel Iruskieta, Mikel Lersundi, Elisabete Pociello, Larraitz Uria, Ruben Urizar

IXA NLP Group, University of the Basque Country (UPV/EHU)
Manuel Lardizabal 1 48014 Donostia
josemaria.arriola@ehu.es

Resumen: En este artículo describimos los experimentos llevados a cabo partiendo de una Gramática de Restricciones en CG-2 para el procesamiento de las posposiciones complejas del *Euskera.* Presentamos el desarrollo y la evaluación de la gramática reescrita en CG-2 y la nueva gramática en CG-3 para el procesamiento de las posposiciones complejas.
Palabras clave: sintaxis superficial, reutilización de recursos lingüísticos.

Abstract: In this paper we describe some experiments based on a previous Constraint Grammar (CG-2) of Basque Complex Postpositions. We present the development and the evaluation of the rewriten CG-2 and the new CG-3 grammars for processing Basque Complex Postpositions.
Keywords: surface syntax**,** reusability of linguistic resources.

1 Introduction

This paper describes some aspects of the design of a rule based grammar for processing Basque complex postpositions, which is built using Constraint Grammar based rules (CG) (Karlsson et al., 1995) that have been implemented by means of CG-2 (Tapanainen, 1996) and CG-3[1]. CG is the formalism or the methodological environment for NLP processing. There have been different implementations or different CG systems such as CG-2 or CG-3, but all of them have in common the following feature: the context-dependent manipulation of tag-encoded linguistic information at the token level.

The primary aim of this exploratory work is to account for as many of the complex postpositions phenomena of Basque as possible and to decide on an initial style guide for the partial syntactic annotation of these phenomena. In developing the rule based grammar for the first time, deciding what constitutes a complex postposition, and how it should be annotated, account for a major part of the previous work (Aduriz et al., 2008). Based on this work, there is a chunking grammar for postpositions. The main idea of this work is to profit the existing CG grammar for recognizing complex postpositions as chunks[2].

Different experiments have been carried out in order to improve the linguistic analysis and finally, we made the decision to convert the CG-2 rules for recognizing chunks to an equivalent CG-3 grammar where instead of tagging explicitly the postposition chunk, we assign the corresponding syntactic function tags. Currently, postpositions chunks are recognized and in addition, the syntactic

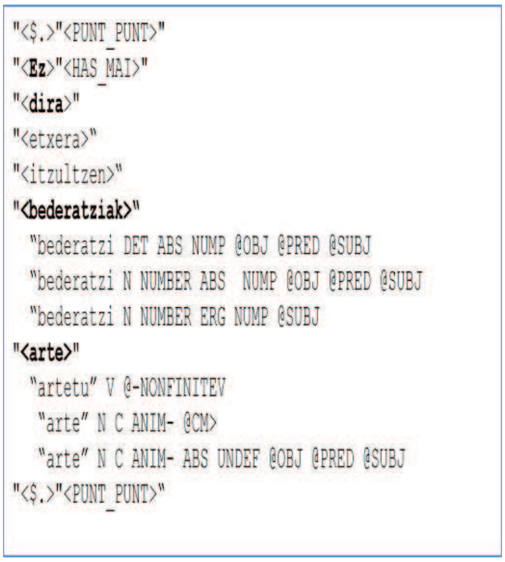
[1] VISL CG-3 Disambiguator version 0.9.7.7873 Copyright (C) 2007-2011 GrammarSoft ApS. All Rights Reserved.

[2] A chunk is a non-recursive phrase (noun phrase, prepositional phrase, verbal chain, etc.) which expresses a constituent (Abney, 1991; Civit, 2003).

function tags that should be given to postpositions have been assigned.

The sentence *Ez dira etxera itzultzen bederatziak arte* ("They don´t come back home until nine o´clock") will be used to illustrate the main steps of the methodology. Figure 1 shows the surface analysis of the sentence before being disambiguated at morphosyntactic level and the phrase marked up in bold face ***bederatziak arte*** (stands for: "until nine o´clock") is the complex postposition structure that will be analyzed in further steps.

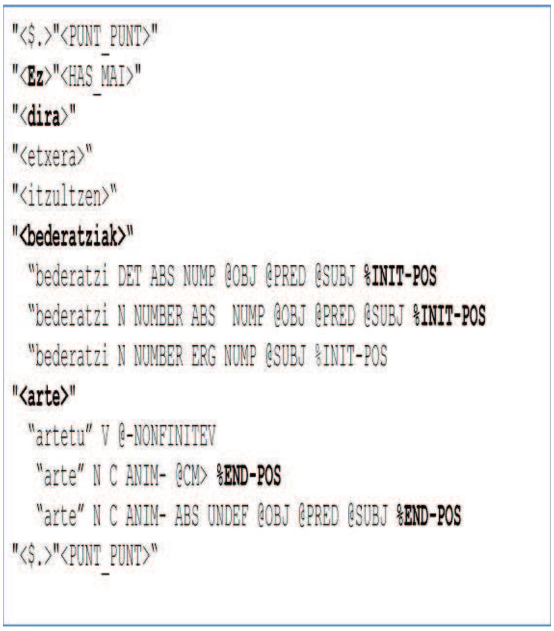
The morphological analyzer (MORFEUS) (Aduriz et al., 1999) assigns multiple possible readings to tokenized input, where every word form is associated with one or more reading lines providing PoS and other linguistic categories. For instance, the token *arte* in the example cohort[3] below presents noun (N) and verb (V) readings. The readings contain morphosyntactic information and CG style syntactic function tags. The syntactic function tags are preceded by the symbol '@'. For instance, *arte* as noun has two possible interpretations: modifier of the word containing the case marker (@CM>) or object (@OBJ), subject (@SUBJ) or predicate (@PRED). Besides, as verb it has the @-NONFINITEV tag that means non-finite auxiliary verb.



```
"<$.>"<PUNT_PUNT>"
"<Ez>"<HAS_MAI>"
"<dira>"
"<etxera>"
"<itzultzen>"
"<bederatziak>"
  "bederatzi DET ABS NUMP @OBJ @PRED @SUBJ
  "bederatzi N NUMBER ABS  NUMP @OBJ @PRED @SUBJ
  "bederatzi N NUMBER ERG NUMP @SUBJ
"<arte>"
  "artetu" V @-NONFINITEV
   "arte" N C ANIM- @CM>
   "arte" N C ANIM- ABS UNDEF @OBJ @PRED @SUBJ
"<$.>"<PUNT_PUNT>"
```

Figure 1: Surface syntax example.

The chunking grammar for dealing with complex postpositions assign the tag %INIT-POS to the first element of the complex postposition and the tag %END-POS to the final element of the postposition. See Figure 2:



```
"<$.>"<PUNT_PUNT>"
"<Ez>"<HAS_MAI>"
"<dira>"
"<etxera>"
"<itzultzen>"
"<bederatziak>"
  "bederatzi DET ABS NUMP @OBJ @PRED @SUBJ %INIT-POS
  "bederatzi N NUMBER ABS  NUMP @OBJ @PRED @SUBJ %INIT-POS
  "bederatzi N NUMBER ERG NUMP @SUBJ %INIT-POS
"<arte>"
  "artetu" V @-NONFINITEV
   "arte" N C ANIM- @CM> %END-POS
   "arte" N C ANIM- ABS UNDEF @OBJ @PRED @SUBJ %END-POS
"<$.>"<PUNT_PUNT>"
```

Figure 2: Chunk postposition example.

The chunking grammar for delimiting complex postpositions works on the output of the morphological analyzer before morphosyntactic disambiguation. It works over the output of the morphological analyzer in a cascaded way. As a result, the complex postposition in Figure 2 has been recognized, but there are many morphological readings for the postposition. Besides, the syntactic function tags are not adequate for a postposition structure. Section 2 deals with Basque complex postpositions. Section 3 describes the experiments that have been carried out in order to improve the surface analysis of complex postpositions. Section 4 presents the evaluation of those experiments. Finally, some conclusions and future work are outlined in the last section.

2 *Basque Complex Postpositions*

Basque is a non-Indo-European language spoken on both sides of the western Pyrenees, is an agglutinative language with a medium to large sized system of affixed case markers/postpositions: 16 affixed cases/postpositions (ergative, absolutive, possessive genitive, local genitive, dative, allative, ablative, inessive, destinative, partitive,

[3] The group of readings for each token is called a *cohort* and the readings manipulated by the operations are called *targets*.

prolative, instrumental, sociative, motivative, directional and terminative[4]).

Lemma2	Suffix1	Suffix2	Examples
arte *(noun)*	-en (genitive)	-an/-ra/-tik/-ko (inessive/alative/ablative/genitive)	Gazteen artean (among young people)
	-0 (no case)	-an/-ra/-tik/-ko (inessive/alative/ablative/genitive)	Jende artean (among people)
	-absolutive	-0/ko (no case)/genitive	Bederatziak arte (until nine)
	-ra (alative)	-0/ko (no case)/genitive	Bilbora arteko trena (the train to Bilbao)
	-0 (no case)	-0 (no case)	Bihar arte (until tomorrow)

Table 1. Complex postpositions for *arte*

The shallow syntactic process is composed of a number of different grammars dealing with chunking, understood as recognition of phrase boundaries (Abney, 1991). We are not concerned with the theoretical discussion of the definition of postpositions. Instead, we prefer to take a practical view in order to improve the shallow syntactic analysis of postpositions. Postpositions in *Basque* play a role similar to that of prepositions in languages like English or Spanish, so that postpositions suffixes are attached to the last element of the phrase. They are defined as "forms that represent grammatical relations among phrases appearing in a sentence" (Euskaltzaindia, 1994). There are two main types of postpositions in Basque: (1) a suffix appended to a lemma and, (2) a suffix followed by a lemma (main element) that can also be inflected.

(1) *teilatu-tik*
roof - (from the)
from the roof

(2) *teilatu-aren gain-etik*
roof-(of the) top-(from the)
from the top of the roof

The last type of elements has been termed as complex postposition. This term is used to name the whole sequence of two words involved, and not just to refer to the second element. Complex postpositions can be described as:

(3) lemma1 + (suffix1 + lemma2 + suffix2)

In these constructions, the second lemma is fixed for each postposition, while the first lemma allows for much more variation, ranging from every noun to some specific semantic classes.

The above description (3) is intended to stress (with parentheses) the fact that the combination of both suffixes with the second lemma acts as a complex case-suffix that is "appended" to the first lemma. Both suffixes present different combinations of number and case, which can agree in several ways, depending on the lemma, case or contextual factors. Table 1 shows the different variants of the complex postposition, derived from the lemma *arte*, which is polysemous (it means i) olm oak, ii) art, iii) time, iv) skill, v) among, vi) until).

These lemmas can be included in the lexicon with the part of speech postposition. This approach is as valid as the one that we have used in the shallow syntactic process. The postposition part of speech has been excluded from the lexicon (EDBL[5]) just to simplify the disambiguation process[6]. Most of the syntactic information is first introduced with all ambiguity regardless of the context and later

[4] The definition of the terms case/postposition is disputed. In the current terminology only ergative, absolutive and dative are considered cases, while the others are considered affixed postpositions.

[5] Currently, the Basque Lexical Database (EDBL) resides under the ORACLE DBMS, on UNIX, and it may be consulted via the Internet (http://ixa2.si.ehu.es/edbl).

[6] After morphosyntactic analysis the tagger/lemmatiser EUSTAGGER (Aduriz, I. and A. Díaz de Ilarraza. (2003) obtains the lemma and category of each form and also performs disambiguation using the part of speech (POS).

select and remove rules take care of disambiguation.

Those postposition structures that are formed by a suffix followed by a lemma (postposition) and that can be also inflected are treated at surface syntactic level as we illustrate in the examples i) *Bederatziak arte* and ii) *Zuhaitz ederren artean*.

(i) *Bederatziak* *arte*

Bederatzi + ak *arte*
(nine) *(until)*

(ii) *Zuhaitz* *ederren* *artean*

Zuhaitz *eder +* *en* *arte +* *an*
(tree) *(beautiful + of)* *(between + in)*

The above postpositions can be analyzed syntactically as postposition structures: the postposition element *arte* in the first example (i) takes as first component an NP in absolutive case *–ak* (*Bederatziak*, stands for "nine"), and in the second example (ii), *arte* is inflected in inessive case *–n* and takes as first component a noun group in genitive case *–en* (*zuhaitz ederren*, stands for "of beautiful tree"). If we look closely at the lexical items that fill in the postposition role, we notice that *arte* is a noun following the information provided by our morphological analyzer.

Most of the main elements of postposition structures are nouns. Literature on *Basque* regarding these elements considers them apostposition part of speech. For instance, (Hualde, 2002) points out some reasons for that: since regular nouns cannot take inflectionless complements, this property would seem to justify treating these elements as postpositions (i.e. as having acquired some properties that distinguish them from nouns).

In this paper, we simply wish to describe the processing of complex postpositions at syntactic level based on the Constraint Grammar (CG) formalism and we will examine the influence of such approach in disambiguation.

The lexical database for Basque EDBL is the main base for the automatic processing of Basque; it includes morphological, syntactic and semantic information. Concerning postpositions, and as it has been mentioned above, these elements are not included in EDBL as lexical entries with postposition category. We opted for annotating these structures as postpositions at shallow syntactic level rather than considering them postposition categories. There are all the lemmas and suffixes that take part in postposition structures in EDBL. As a result, most of the main elements (lemmas) of these structures are included in EDBL with the noun part of speech and the corresponding syntactic function tags[7] for nouns:

(iii) *Bederatziak*
DET @OBJ @PRED @SUBJ
N @OBJ @PRED @SUBJ
N @SUBJ

arte
V NON-FINITE @NON-FINITEV
N C ANIM- @OBJ @PRED @SUBJ
N C ANIM- @CM>

The example (iii) shows that the postposition *arte* has four syntactic function tags corresponding to a noun and one for the non-finite verb interpretation. Besides, the first element of the postposition structure *bederatziak* has seven syntactic function tags taking into account the different morphological analysis. Therefore, the ambiguity rate is quite high. In Basque, both morphological and syntactic ambiguity exists, i.e. one word receives multiple analyses. Morphological ambiguity in Basque includes part of speech ambiguity e.g. typically noun/verb ambiguity. For agglutinative languages there are additional sources of ambiguity (number, case, etc.). Syntactic ambiguity is added on top of morphological ambiguity.

The grammar developed in this experiment will assign the appropriate syntactic function tag to all the elements of the postposition structure and it will reduce the ambiguity in the following way: one syntactic tag for *bederatziak* and one adverbial syntactic tag for *arte*:

(iv) *Bederatziak* *arte*
DET @CM> N @ADVERBIAL

[7] Main syntactic function tags: subject (@SUBJ), object (@OBJ) and predicate (@PRED). Modifier function tags: @CM> stands for modifier of the element carrying case. Functions related with verbs: @-NON-FINITEV.

As we can see in (iv), the analysis obtained after applying the grammar rules shows a postposition structure with adverbial function.

The new syntactic function tags attached to the complex postpositions and the reduction of the ambiguity reverts in the improvement in the quality of several applications, such as Part-Of-Speech (POS) taggers and parsers. However, our study is restricted to shallow syntax since we could not deal with constructions that are semantically and syntactically ambiguous. Morphosyntactic information is insufficient to take care of this kind of ambiguity.

3 *Experiments*

Two experiments were carried out in order to compare both approaches: the first one, reusing the CG-2 chunking grammar containing the mapping rules for recognizing the complex postpositions, and the second one, reformulating only the linguistic information contained in the CG-2 rules by means of CG-3 new rules. Both grammars were applied to a test corpus (1680 tokens) that contains 70 postposition cases in order to find discrepancies on the results.

REPLACE works like a mapping operator, closing the line for further mapping. It is less versatile than SUBSTITUTE, but backward compatible with CG-2. Substituted tags can be "seen" by later SUBSTITUTE or MAPPING rules, even in the same section. Usually as a special section (CORRECTIONS or BEFORE-SECTIONS), but in CG-3 are allowed anywhere.

3.1 Reusing CG-2 mapping rules

The grammar developed in this experiment will assign the appropriate syntactic function tags to all the elements of the postposition structure and it will reduce the ambiguity. For that purpose, a set of REPLACE[8] rules was written in order to attach to the complex postposition components the corresponding surface syntactic tag. Those REPLACE rules have to take into account the syntactic function tag corresponding to the lexical part of the complex postposition and the syntactic function tag to the word that takes the case demanded by the main postposition element. The rules make use of the morphosyntactic information and the chunk tags assigned by the chunker. For instance, we work on the example *bederatziak arte*. The REPLACE rules just change the syntactic function tag, but they maintain the morphological information as well as its corresponding chunk marker tag that indicates the initial part and the ending of the complex postposition. The REPLACE rules to assign the appropriate syntactic function tags to the words of the complex postposition *bederatziak arte* have the following format:

```
REPLACE (N C ANIM- ABSOLUTIVE MG
      @ADLG %FIN-POS56)
     TARGET (N %END-POS56)
     IF (-1 (%INIT-POS56));
```

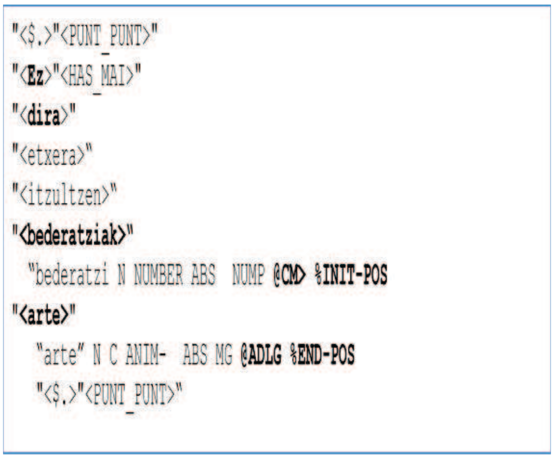
The above REPLACE rule attaches to the postposition *arte* the adverbial function tag (@ADLG, adverbial complement) and maintains its morphological analysis as well as its corresponding chunk marker tag (%END-POS56). On the other hand, the ambiguity between the noun and the verb reading of *arte* is resolved making use of the chunk marker tag and the noun reading is selected by means of the following rule:

```
REMOVE (VERB) IF (0 NOUN)
   (-1 ({POS-HAS56));
```

With regard the postposition element *bederatziak* the following REPLACE rule attaches case marked element modifier function tag (@CM>) and maintains its morphological analysis as well as its corresponding chunk marker tag (%INIT-POS56).

```
REPLACE (DET DZH NMGP ABS NUMP MUGM
       @CM> %INIT-POS56)
     TARGET (DET %INIT-POS56))
      IF (1 ((%END-POS56));
```

As a result of applying those rules, we get the following analysis in Figure 3:



```
"<$.>"<PUNT_PUNT>"
"<Ez>"<HAS_MAI>"
"<dira>"
"<etxera>"
"<itzultzen>"
"<bederatziak>"
  "bederatzi N NUMBER ABS  NUMP @CM> %INIT-POS
"<arte>"
   "arte" N C ANIM-  ABS MG @ADLG %END-POS
   "<$.>"<PUNT_PUNT>"
```

Figure 3: Postposition after applying REPLACE rules.

[8] REPLACE is a CG-2 operator retained in Vislcg and CG-3 (http://beta.visl.sdu.dk/).

The grammar is composed of 74 REPLACE rules and 14 disambiguation constraint rules. One set of REPLACE rules assigns to the initial part of the postposition the following syntactic function tags: @CM> that stands for modifier of the element carrying case or @NOUN-COMPLEMENT that stands for the noun modifier. Another set of rules replaces the syntactic function tag of the final element of the postposition with the function tag @ADLG that stands for adverbial function tag.

The rules refer to the initial element of the complex postposition and therefore apply to any word that is tagged with %INIT-POS as well as the characteristics that appear in context specification of the rule in order to replace the syntactic function tag with @CM> or @NOUN-COMPLEMENT. In the same way, other set of rules refer to the final element of the complex postposition and therefore apply to any word that is tagged with %END-POS and the characteristics that appear in context specification of the rule in order to replace the syntactic function tag with the adverbial @ADLG.

The corpus used to develop the grammar contains 53,324 tokens and the ambiguity rate is 5.46 % analyses per token. After applying the grammar, the ambiguity rate per token is 5.3%. In other words, 7,665 syntactic function tags that are inadequate have been removed.

3.2 Defining a CG-3 grammar

The idea is to avoid the amount of different tags, such as tags for recognizing the postpositions as chunks, tags for recognizing another kind of chunks and so on. The idea is also to integrate syntactic mapping with the morphological disambiguation. For this purpose, new CG rules were written by means of the type of rule SUBSTITUTE. This module is composed by 150 SUBSTITUE rules.

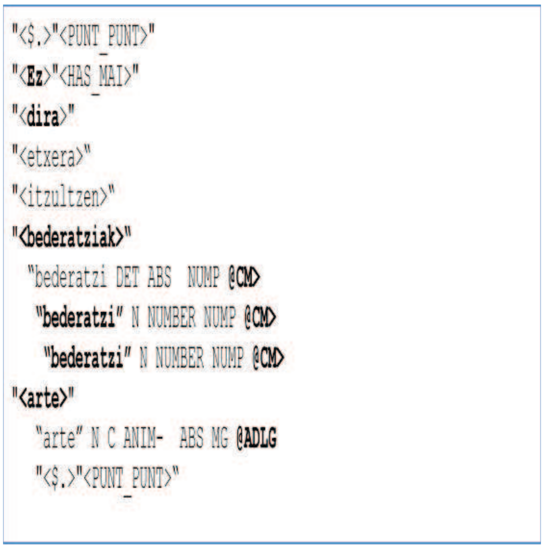
The rules make use of the morphosyntactic information and the chunk tags attached by the chunker in an implicit way. In this case, instead of adding a tag for the initial part and the ending part of the complex postpositions, the elements of the complex postpositions are tagged with the syntactic function tag corresponding to this structure. For instance, for the first element of the postposition *bederatziak* the following SUBSTITUTE rule is applied:

```
SUBSTITUTE (@PRED @OBJ @SUBJ) (@CM>)
   TARGET IZE-DET-IOR-ADJ-ELI-SIG
      IF (0 ABS + MUGATUA)
      (1 POST-56IZE +IZE_ABS_MG) ;
```

The SUBSTITUTE rule for the postposition *arte* is also based on the morphosyntactic information and in the previously defined postoposition tagsets:

```
SUBSTITUTE (@PRED @OBJ @SUBJ)
(@ADLG) TARGET POSTPOSIZIOAK-5 IF (-1
   IZE-DET-IOR-ADJ-ELI-SIG + ABS +
           MUGATUA) ;
```

The main idea in both SUBSTITUTE rules is to substitute the syntactic function tag that is assigned by the morphological analyzer, because these syntactic function tags are not adequate for complex postpositions. As a result of applying those rules, the following analysis is obtained:



```
"<$.>"<PUNT_PUNT>"
"<Ez>"<HAS_MAI>"
"<dira>"
"<etxera>"
"<itzultzen>"
"<bederatziak>"
 "bederatzi DET ABS  NUMP @CM>
  "bederatzi" N NUMBER NUMP @CM>
   "bederatzi" N NUMBER NUMP @CM>
"<arte>"
  "arte" N C ANIM-  ABS MG @ADLG
  "<$.>"<PUNT_PUNT>"
```

Figure 4: Postposition after applying CG-3 SUBSTITUTE rules.

In the analysis of *arte* two interpretations have been discarded by means of some basic disambiguation rules. But the section for recognizing complex postpositions in an implicit way has replaced the previous existing syntactic function tags assigned to a noun with the syntactic function tag corresponding to the postposition (@ADLG).

4 *Evaluation*

The test corpus (1680 tokens) is a running text and contains 70 postposition cases. The postpositions were annotated by the chunker on the analysis of the morphological analyzer. This leads, on the one hand, to a high ambiguity rate, but, on the other hand, it takes into account all the morphological analyses that are necessary for chunking. The evaluation is divided into two parts: the first one deals with the grammar composed by CG-2 REPLACE rules and the second one, with the CG-3 grammar of SUBSTITUTE rules.

4.1 Evaluation of CG-2 REPLACE rules

These postpositions were annotated with the appropriate syntactic function tag by means of 74 REPLACE rules. The syntactic ambiguity rate per token before applying the rules was 5.97 syntactic analyses per token. After applying the rules, the ambiguity rate is 5.71 analyses per token. In 5 of the 70 postposition cases the existing chunking rules do not manage to annotate the postposition chunk correctly. Wrong applications of the rules are mainly due to the high ambiguity of some words and scope mistakes of the rules. The precision[9] and recall[10] for the REPLACE rules involved in the correct assignment of syntactic functions and disambiguation are 94.2% and 100% respectively. In the four cases, the REPLACE rules need to be refined in order to apply the correct syntactic function tag.

Taking into account the general impact of the annotation of complex postpositions on the overall analysis with respect to the qualitatively complexity of postpositions, we think that this approach is necessary in order to improve syntactic disambiguation, because they are distributed across two words, and they also show different kinds of syntactic agreement.

4.2 Evaluation of CG-3 SUBSTITUTE rules

In this case, all the tested postpositions were annotated with the appropriate syntactic function tag by means of 150 SUBSTITUTE rules. This is due to the fact that we have learnt from the mistakes of the previous experiment. When defining the section of SUBSTITUTE rules, we find two main differences with respect the REPLACE section of CG-2: 1) the SUBSTITUTE rules do not maintain the tag for the initial and ending part of the complex postposition and 2) the SUBSTITUTE rules are integrated into a general disambiguation grammar. The REPLACE rules are in an independent grammar just for dealing with complex postpositions. Nevertheless, in both cases, the aim of those rules is to prepare the input for further steps: first, for morphosyntactic disambiguation and secondly, for syntactic disambiguation.

In this approach, the complex postpositions have not been marked up in an explicit way, but they have been recognized by means of the appropriate syntactic function tags attached to the elements that compose the complex postposition structure. The idea is to deal with them when processing more general chunks.

5 Discussion and future work

Integrating chunk mapping with morphological disambiguation is problematic, since the mapping rules will overmap in the presence of multi-ambiguous morphological cohorts. In order to reduce the ambiguity, two experiments have been carried out. In the first one, the CG-2 grammar composed of REPLACE rules has been applied in order to assign the correct syntactic function tag to the elements that form the complex postposition structures. As a consequence of the overmapping, there are more ambiguous contexts to apply the REPLACE rules. Therefore, we have to go through rules and add a lot of C´s (safety contexts) to existing contexts, and/or add NOT contexts. Nevertheless, as we have seen in the evaluation the results are satisfactory from a linguistic point of view, because the postpositions get the adequate syntactic function tag.

Rewriting the content of the CG-2 rules by means of CG-3 is another way of solving the ambiguity of postpositions. In that case, we prefer to assign to those elements that form part of the complex postposition just the correct syntactic function tag. In this approach, these structures have not been marked up in an explicit way, but they have been recognized implicitly. The idea is to deal with them when processing more general chunks. In order to distinguish postpositions, another alternative would be to add the analysis of the part of

[9] precision = correctly detected postpositions/(correctly detected postpositions + wrong postpositions)

[10] recall = correctly detected postpositions/all postpositions

speech postposition to the main element of complex postpositions.

Finally, in future projects we plan to improve the grammar for recognizing more complex postpositions.

Ackowledgments

This research was supported by the Basque Government (IT344-10). Jose Mari Arriola's experiments were done during the research stage in Helsinki (funded by the Basque Government).

Bibliografía.

Abney, S. 1991. *Parsing by Chunks*. In Robert C. Berwick, Steven P. Abney, and Carol Tenny, editors, *Principle-Based Parsing: Computation and Psycholinguistics*, pages 257-278. Kluwer Academic Publishers, Boston.

Aduriz I., Aldezabal I., Aranzabe M. J., Arriola J. M:, Ceberio K., Estarrona A., Iruskieta M., Lersundi M., Pociello E., Uria L., Urizar R., Aldasoro E. 2008. *Euskarazko postposizio-lokuzioen tratamendu konputazionala*. UPV / EHU LSI / TR 07-2008.

Aduriz, I. and A. Díaz de Ilarraza. 2003. Morphosyntactic disambiguation and shallow parsing in Computational Processing of Basque. *In Inquiries into the lexicon-syntax relations in Basque*. Bernarrd Oyharabal (Ed.), Bilbao.

Aduriz I., Arriola J., Artola X., Díaz de Ilarraza A., Gojenola K., Maritxalar M., Urkia M. 2000. *Euskararako murriztapen-gramatika: mapaketak, erregela morfosintaktikoak eta sintaktikoak*. UPV/EHU/LSI/TR12-2000.

Aduriz I., Agirre E., Aldezabal I., Arregi X., Arriola J., Artola X., Gojenola K., Sarasola K., Urkia M. 1999. *MORFEUS: Euskararako analizatzaile morfosintaktikoa*. UPV/EHU/LSI/TR 1-99.

Bick, E. 2000. *The Parsing System Palavras - Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Famework*, Aarhus: Aarhus University Press

Civit, M. 2003. *Criterios de etiquetacion y desambiguacion morfosintactica de corpus en Español*. PhD thesis, Universidad de Barcelona.

Euskaltzaindia 1994. *Euskal Gramatika: Lehen Urratsak-I*. Euskaltzaindia.

Hualde J. I. 2002. Regarding Basque postpositions and related matters. *In Erramu Boneta: A Festschrift for Rudolf P.G. de Rijk*, ed. by Xabier Artiagoitia, Patxi Goenaga & Joseba Lakarra, p. 325-339. Bilbao: Univ. Del PaísVasco/Euskal Herriko Unib. Supplements of ASJU 44.

Karlsson, F.,Voutilainen A., Heikkilä J., Anttila A.. 1995. *Constraint grammar: A language-independent system for parsing unrestricted text*. Berlin & NewYork: Mouton de Gruyter.

Tapanainen, P. 1996. *The Constraint Grammar Parser CG-2.* University of Helsinki Publications No. 27.