Different Issues in the Design and Implementation of a Rule Based Grammar for the Surface Syntactic Disambiguation of Basque

Jose Mari Arriola Egurrola
UPV/EHU University of the Basque Country
josemaria.arriola@ehu.eus

Abstract

This paper presents the results of a set of preliminary experiments reusing some of the modules for the surface syntactic processing of Basque in order to improve the surface syntactic disambiguation. The general idea is to reuse the existing modules at different stages of processing and to find the better order of application of those modules. It aims at introducing a strategy for surface syntactic disambiguation in Basque via rule-based grammars. The results from an evaluation of this disambiguation strategy on a sample corpus are described.

1 Introduction

We will describe some practical issues raised during the design of the strategy for a rule based grammar implemented by means of VISL CG3 (Didriksen, 2010). This work is undertaken in the frame of the Constraint Grammar formalism (Karlsson et al., 1995), and focuses on the design of a disambiguation module which involves both morphology and syntax. The results of a set of preliminary experiments for the design and evaluation of the rule based grammar are presented in order to improve the surface (Abney, 1997) syntactic disambiguation module.

The general framework for the syntactic processing of Basque is composed by several modules (see Figure 1). The IXA research group[1] is working on a robust parsing scheme that provides syntactic analysis in an incremental fashion. Information contained in the EDBL lexical database for Basque (Aldezabal et al., 1999; 2001) constitutes the basis for our analyzers. Once textual input has been tokenized, morphologically analyzed by Morfeus, (Alegria et al., 1996; 1997) and disambiguated by means of Eustagger (Aduriz et al., 2003), syntactic information is added in three distinct stages of processing: i) a CG grammar assigns the syntactic functions to each word-form and deals with the morphosyntactic ambiguity; ii) a CG disambiguation grammar is applied to disambiguate the syntactic functions; iii) a chunk parser provides a partial constituent analysis; and finally, iv) a dependency parser establishes the dependency links (see Figure 1).

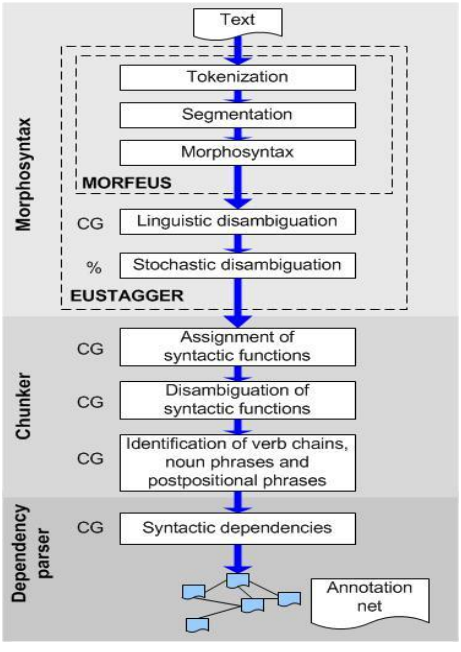

Fig. 1: General framework

We will focus on the first task in the overall parsing scheme, namely the improvement of the syntactic disambiguation process when assigning the syntactic functions. When the syntactic disambiguation grammar module

[1] URL: ixa.si.ehu.es

was first developed, the CG-2 parser (Tapanainen, 1996) was used to implement the rules. There was a main grammar containing morphosyntactic disambiguation rules and purely syntactic disambiguation rules. In a second stage, the main grammar was split into two subgrammars one for morphosyntactic disambiguation and the other for syntactic disambiguation.

Being a language with rich morphology, the basis for the syntactic processing is morphology. The surface oriented syntactic function tags (Karlsson et al., 1995) are assigned by the same module that adds the morphological information to the lemma. The idea is that morphology and syntax are closely related and in a number of cases the syntactic function can be unambiguously tagged to the morphological representation, for instance the ergative plural suffix *–ek* is always subject (@SUBJ). Most of the syntactic information is first introduced with all ambiguities regardless the context and later select and remove rules take care of disambiguation. Both morphological and syntactic ambiguities exist, i.e. one word can receive multiple analyses. Morphological ambiguity includes e. g. part of speech ambiguity e.g. typically noun/verb or noun/adjective. For agglutinative languages there are additional sources of ambiguity (number, case, etc.)

Our aim is to improve the analysis by making use of the information of some of the existing modules, and taking into account that the order of application of the different modules is very important. In this paper we explore a new combination strategy of the modules and respective influence of reordering those modules.

2 Related work

In the literature, we find several approaches to improve syntactic disambiguation. There are three prominent tendencies in disambiguating-grammars and in syntactic analyzers: those based on linguistic descriptions, those based on statistical techniques, and finally, hybrid methods, which combine both. Morphologically rich languages present new challenges, as the use of state of the art parsers for more configurational and non-inflected languages like English does not reach similar performance levels in languages like Basque, Greek or Turkish (Nivre et al., 2007).

We consider the correct morphological disambiguation as a basis for the surface syntactic processing (DoleŽalova, J. and PetkeviČ, V. 2007). In the same direction (Agirre et al, 2012) revealed that the most relevant information is the case carried by the noun and the transitivity of the verb. Besides, (Bengoetxea et al., 2012) shows that POS errors harm the parser.

Ambiguity arises from previously done morphological analyses, and hence, it is closely dependent on decisions made at the morphological level. If only categorial (POS) ambiguity is taken into account, there is an average of 1.55 interpretations per word form, which rises to 2.65 when the full morphosyntactic information is taken into account, giving an overall 64% of ambiguous word-forms. We chose the CG formalism for our purposes of starting to handle syntax. In fact, there are several works that show that good results have been obtained when parsing with CG (Karlsson et al., 1995; Samuelsson and Voutilainen, 1997; Tapanainen and Järvinen, 1997; Bick, 2000).

We have moved from CG-2 to CG3 taking into account the bigger expressive power of CG3 and the open source philosophy. VISL CG3 has been extended to many languages[2].

3 Some experiments

The original CG grammar rules for disambiguation were written in CG-2 and now have been reimplemented and expanded with CG3. In the experiments we explore how much we can improve our syntactic analysis by means of exploiting the interrelation between the different modules. Firstly, some attempts in that direction will be described more specifically in the grammar for morphosyntactic disambiguation. We have included a module for lexical correction for the treatment of complex postpositions and we have tried to benefit from the analysis of the

[2] http://beta.visl.sdu.dk/constraint_grammar_languages.html

chunker. Afterwards, we will focus on the subject/object ambiguity. The subject/object ambiguity in Basque caused by the homonymy of aboslutive plural and ergative singular (approximately 40% of the ambiguity left after morphosyntactic disambiguation). Functional ambiguity between subject and object is a widespread problem in Basque, where 22% of subjects and objects are ambiguous, and this ambiguity surfaces in 33% of the sentences. This problem is comparable to PP attachment ambiguities in other languages (Atutxa et al., 2012).

```
"<$.>"<PUNT_PUNT>"
"<Goizeko>"
  "goiz" ADJ C GEL S @<IZLG @IZLG>
  "goiz" ADJ C GEL S @OBJ @PRED @SUBJ
  "goiz" N C DIS ANIM- @ADLG
  "goiz" N C DES ANIM- @<IZLG @IZLG>
  "goiz" N C ABS ANIM- UNDET @OBJ @PRED @SUBJ
  "goiz" N C GEL ANIM- S @<IZLG @IZLG>
  "goiz" N C ABS ANIM- S @OBJ @PRED @SUBJ
"<bederatziak>"
  "bederatzi" DET ABS PL @OBJ @PRED @SUBJ
  "bederatzi " N NUMBER ABS PL @OBJ @PRED @SUBJ
  "bederatzi " N NUMBER ERG S @SUBJ
"<arte>"
  "artetu" V @NONFINITEV
  "arte"   N  C ANIM-  ABS @OBJ @PRED @SUBJ
  "arte"   N  C ANIM- @CM>
"<ez>"
"<nuen>"
 "*edun" VAUX NR_HURA NK_NIK @+JADLAG
 "*edun" VAUX REL NR_HURA NK_NIK @+JADLAG_MP_IZLG>
 "*edun" VAUX ZHG NR_HURA NK_NIK @+JADLAG_MP_OBJ
 "*edun" VAUX MOS NR_HURA NK_NIK @+JADLAG_MP_ADLG
 "ukan" V NR_HURA NK_NIK @+JADNAG
 "ukan" V REL NR_HURA NK_NIK @+JADNAG_MP_IZLG>
 "ukan" V ZHG NR_HURA NK_NIK @+JADNAG_MP_OBJ
 "ukan" V MOS NR_HURA NK_NIK @+JADNAG_MP_ADL
"<lortu>"
 "lortu" V PART @-JADNAG
 "lortu" V PART BURU  @-JADNAG
 "lortu" V PART ABS MG @-JADNAG_MP_OBJ
         @-JADNAG_MP_PRED @-JADNAG_MP_SUBJ
"<zure>"
"<ezpainak>"
 "ezpain"  N C ABS PL @OBJ @PRED @SUBJ
  "ezpain" N C ERG S @SUBJ
"<ikustea>"
  "ikusi"  V NOUNV KONP ABS @-JADNAG_MP_OBJ
           @-JADNAG_MP_SUBJ
  "ikusi"  N C ANIM- ABS S @OBJ @PRED @SUBJ

"<$.>"<PUNT_PUNT>"
```

Fig. 2: Morphological analysis

We will use the following sentence as a first example to illustrate the main steps of the methodology:

(1) *Goizeko bederatziak arte ez nuen lortu zure ezpainak ikustea* (stands for: Until nine o´clock in the morning I don´t achieve to see your lips).

In Figure 2 we have the morphological analysis of the sentence. We have simplified the analysis and we have concentrated on the phrases marked up in bold face *bederatziak arte* (stands for: until nine o´clock), which is the complex postposition structure that will be analyzed in further steps and *ezpainak* (stands for: lips). The ergative singular/absolutive plural ambiguity can be seen in the word *ezpainak* 'lips'. The form can potentially be a subject, object or predicate in absolutive case and a subject in ergative case. In this sentence it is an object and therefore in absolutive case. The ergative interpretation can be discarded based on valency requirements of the nominalized verb *ikustea* 'seeing'.

3.1 Lexical correction module

In the grammar for morphosyntactic disambiguation we have included a module of lexical correction for the treatment of complex postpositions. Prior to the morphosyntactic and syntactic disambiguation process, as a first step we design a lexical correction module composed by 155 SUBSTITUTE rules in order to eliminate unnecessary syntactic ambiguity. The complex postpositions have been processed at surface syntactic level instead of including these elements in the lexicon. As a result, we have inadequate syntactic tags for those structures.

Postpositions in Basque play a role similar to that of prepositions in languages like English or Spanish, so that, postposition 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). We have treated at the surface syntactic level those postposition structures that are formed by a suffix followed by a lemma (postposition) that can be also inflected: *bederatziak arte* (stands for: until three).

The postposition element *arte* takes, as first component, an NP in absolutive case *–ak* (*bederatziak*, stands for nine). We can see 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. The elements of the complex postposition 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 postposition 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 we have reduced the starting syntactic ambiguity of the first example (see Figure 3):

```
"<bederatziak>"
  "bederatzi" N NUMBER ABS PL @CM>
"<arte>"
    "arte"   N  C ANIM- @ADVERBIAL
```

Fig. 3: Lexical correction of complex postpositions

The word *bederatziak* is holding after lexical correction rules the syntactic function tag @CM> (stands for: case-marker modifier) and *arte* the syntactic function tag @ADVERBIAL (stands for: adverbial). In the following step the grammar for disambiguating the syntactic function tags can select the appropriate function.

3.2 Reutilisation of chunker tags

When working with rich morphology languages like Basque it is crucial to make a basic distinction between disambiguation of main syntactic function tags and modifier syntactic function tags. The main syntactic function tags begin with the @-symbol (e.g. @SUBJ for subject, @OBJ for object, @PRED for predicate, etc.) The modifiers have tags that indicate the direction where the head of the phrase could be found but the modifiers and heads are not formally connected. The modifiers make use of the symbols < or > to indicate the direction in which could be found the head (e.g. @ND> for noun determiner, @NC> for noun complement, @CM> case-marker modifier, etc.)

The basic morphosyntactic disambiguation step deals with the disambiguation between main syntactic function tags and modifier syntactic tags, see figure 4:

```
"<Goizeko>"
  "goiz" N C ANIM- GEL S @IZLG>
```

Fig. 4: Basic morphosyntactic disambiguation

The word *goizeko* (stands for: in the morning) is a noun-complement and the head is on the right (that information is given by the symbol >).

On the output of the morphosyntactic disambiguation we applied one module of the rule-based chunker (RBC henceforth, Aranzabe et al., 2009), the one composed of 479 rules to deal with noun phrases. The chunker delimits the chunks with three tags, using a standard IOB marking style. The first one is to mark the beginning of the phrase (B-VP if it is a verb phrase and B-NP whether it's a noun phrase) and the other one to mark the continuation of the phrase (I-NP or I-VP, meaning that the word is inside an NP or VP). The last tag marks words that are outside a chunk.

In order to illustrate the reutilisation of the tags attached by the chunker, we will use the following example:

(2) *Microsoft etxea Word ari da euskaratzen* (stands for: Microsoft is translating Word into Basque).

After applying the rule-based chunker we get partial constituent analysis:

[3] **%SIH**[**Microsoft** etxea]**%SIB** [**Word**]**%SINT** [ari da euskaratzen]

We will focus on the NPs *Microsoft etxea* and *Word*. The NP *Microsoft etxea* is

[3] %SIH: initial part of a phrase; %SIB: ending part of a phrase and %SINT: phrase.

composed by the modifier *Microsoft* and the head *etxea.* In this case we base on the %SIH tag added by the chunker to discard the main syntactic function interpretations:

```
SELECT: (ZERO) IF (0C IZE LINK  0 (%SIH))
(1C (%SIB));
```

Regarding the NP *Word,* in this case it is constituted by one element, it is an independent phrase. We base on the %SINT tag to discard the modifier syntactic function tags:

```
REMOVE:kendu_zeroa, (ZERO) IF (0C
(%SINT));
```

3.3 Verb valency

There have been many attempts to include the subcategorization information in NLP parsers. Bick (2000) uses syntactic verb valency tags specifying e.g. transitivity and selection preferences for various NLP tasks. The use of verb valency is on a high level of grammatical analysis and requires a number of other linguistic resources. Bick (2000) uses tags specifying transitivity preferences such as <vt-vi>, meaning "preferably transitive, but potentially intransitive", but also selection preferences as in the entry for the Portuguese verb convidamos <+ACC-hum> 'invite', where the accusative needs to be a noun denoting a human.

Wiechetek and Arriola (2011) worked on applying verb valency information in the syntactic disambiguation process and they demonstrated that it is convenient.

We think that this kind of information should be included after some basic morphosyntactic disambiguation tasks have been done. It is clear that in some cases pure morphosyntactic information is not enough to solve the syntactic ambiguities. We have included by means of CG mapping rules the valency information developed in the work Building the Basque PropBank (Izaskun et al., 2010) into the CG grammar. In the previous version there was detailed subcategorization information for the most 100 used verbs.

We base on Aldezabal et al. (2010) converting this verb information into valency tags. Each verb can have several frames of argument constellations. In order to simplify the example we will take the verb *lortu* (stands for: to achieve, get), which has only one frame:

lortu V Agcase_ERG Agsyn_Subj Agsem_Human Thcase_Abs Thsyn_Obj

Arguments are ordered by semantic roles (e.g. agent, theme, topic, patient, location) because they are more unique than syntactic arguments (it is very common to have several adverbials in one sentence).

The semantic role level is furthermore perceived as being more abstract and therefore more language independent, which makes it suitable for reuse for other languages. Arguments have 3 possible attributes: case (or postposition) such as (nominative, accusative, ergative), syntactic function (subject, object, adverbial), and selection restrictions (human, concrete, place).

In the case of the verb lortu 'to achieve', the first argument, characterized by the semantic role agent, has the three attributes Agcase_ERG (ergative case) Agsyn_Subj (syntactic function subject) and Agsem_Human (selection restriction human).

The annotation of valency by means of Constraint Grammar rules has the following format adding the valency tags to the verb *lortu* 'achieve, get':

```
ADD (Agcase_Erg Agsyn_Subj Agsem_Human
Thcase_Abs Thsyn_Obj)

TARGET (ADI) IF (0 LORTU);
```

This format is sufficient for the annotation of a small amount of verbs for testing purposes. For a large-scale annotation of verbs we would like a verb database to be the basis from which tags are automatically induced.

Concerning the ambiguity of *ezpainak* (stands for: lips), the ergartive interpretation can be discarded based on valency requirements of the nominalized verb *ikustea* (to see). In this case we can´t make use of one important source of information for the disambiguation that is the agreement between the finite verb and the auxiliary.

When we deal with non-finite verbs, another strategy is to attach the verb auxiliary

information to non-finite verb forms in order to exploit this information when the auxiliary is elided. We make use of the following tags:

- DU: attached to those verbs that allow an auxiliary for transitive verbs, e.g. *lortu*.
- DA: attached to those verbs that allow an auxiliary for intransitive verbs, e.g. *loratu*.

However, in the illustrative example (1) that we have used for describing the methodology we can´t solve the syntactic ambiguity of *ezpainak* even with the auxiliary tag information.

Both morphosyntactic and syntactic ambiguity are tightly related. For that reason, in the first step we deal with morphosyntactic ambiguity and in the second step we deal with syntactic ambiguity. Our strategy to obtain the best disambiguation option was to do the morphosyntactic disambiguation first, and then once we have selected the absolutive or ergative case option we will deal with the syntactic ambiguity.

Another constraint grammar module contains disambiguation rules that make use of the valency. In the case of *lortu* 'achieve, get' in example (1), the ambiguity between the predicative and the object reading of *ezpainak* 'lips' is resolved by means of the valency of ikusi 'to see' and the object reading is selected by means of the following rule:

```
SELECT (@OBJ) IF (0 ABS LINK 0 OBJ) (NOT 0 ERG) (*1 Thcase_Abs LINK 0 Thsyn_Obj BARRIER ADI/ADL/ADT);
```

The other ambiguity in the sentence consists in the readings of the non-finite verbal noun *ikustea* 'seeing', which can be a subject, an object or a predicate. In order to select the object reading the rule checks if there is a verb, here *lortu* 'to achieve' to its left, that has an object in its valency:

```
SELECT (@-NON-FINITEVERB_CLAUSE_OBJ) IF (O NON-FINITE-VERB) (*-1 Thsyn_Obj BARRIER ADI/ADL/ADT);
```

4 Evaluation

The test corpus is divided into two parts: one for developing the grammar and the other one for testing the grammar on unseen corpus (53.324 tokens). We have tested the three basic experiments described in section 3. In the two first tests we have taken into account apart from POS and subcategory all the morphosyntactic information (case, number, type of subordinate clause, etc.). In the third one we tested the syntactic disambiguation and finally for verb valency we have used a smaller sample.

4.1 Lexical Correction module for Complex postpositions

We present firstly the figures of the morphosyntactic disambiguation grammar without the lexical correction module.

Words	**R**	**P**	**F**
Standard	91.40	71.92	80.50
Non-Standard	86.17	58.28	69.53
Out of Lexicon	81.06	36.08	49.93
Total	91.00	69.62	78.89
Total + PM	92.63	74.03	82.30

Table **1**: Initial results

(R=recall; P= precision; F= f-score; PM= puntuaction marks)

After applying the disambiguation grammar with the lexical correction module we obtain a moderate improvement of the results.

Words	R	P	F
Standard	92.42	72.58	81.19
Non-Standard	89.53	60.27	72.05
Out of Lexicon	77.80	36.45	49.64
Total	91.59	70.38	79.59
Total + PM	93.12	74.79	82.90

Table **2**: Lexical correction module effects

We have tested on a smaller sample (233 words) that doesn´t contain non-standard words. Applying the same grammar we obtain better results, recall 94.93 and precision 76.78.

These results show that the effect of the non-standard words on the disambiguation of the standard words should be taken into account when designing the grammar.

4.2 Chunker tags reutilisation

Testing at morphosyntactic level the effect of combining the morphosyntactic disambiguation grammar and the information of the chunker we don´t get better results.

Words	R	P	F
Standard	91.42	73.49	81.48
Non-Standard	88.66	61.00	72.27
Out of Lexicon	72.67	37.85	49.78
Total	90.73	71.47	79.96
Total + PM	92.41	75.72	83.24

Table **3**: Chunker tags reutilisation effects

In the next section we will show the effects on the disambiguation of syntactic function tags.

When analyzing the results shown in Table 2 and Table 3 we should take into account some features of the disambiguation process:

- Complexity of some ambiguities: case (need to deal with the agreement of verbs with subject, object, etc.); the type of subordinate sentence in auxiliary verbs; elided or non-elided element in auxiliary verbs; relative clause versus past tense verb; etc.
- Treatment of variants
- Some errors of the chunker
- Some divergences with the Gold Standard

4.3 Combining lexical correction module and chunker tags

The syntactic ambiguity rate of the testing sample is 5.469 syntactic tags per word. We have tested the effect of combining different modules.

	Input	Output1	Output2	Output3
Anal./Token	5.469	1.930	1.260	1.217
Error rate		4.17	8.6	9.2

Table **4**: Syntactic disambiguation results

The first results (Output1) have been obtained applying just the morphosyntactic disambiguation grammar, the following results for the second output (Output2) have been obtained by means of the disambiguation grammar for syntactic functions and the third one (Output3) has been obtained applying the grammars in this way: first the morphosyntactic disambiguation grammar, secondly on the output of the morphosyntactic grammar the chunker and finally on the output of the chunker the syntactic function disambiguation grammar.

In Table 4 we can see that as the CG-based syntactic disambiguation grammars and the chunker are applied after morphological processing, the errors are propagated and augmented.

4.4 Verb valency

In many cases, we have seen that pure syntactic information is not sufficient for the morpho-syntactic disambiguation of nouns, and richer linguistic information is needed. For instance, in example (1) we have seen that we need verb valency information to solve the ambiguity of *ezpainak*.

The test cases used in this experiment regard morphosyntactic disambiguation; especially we will focus on the ambiguity of the suffix *–ak*. The test includes a set of 10 verbs for disambiguation. In the text that we have chosen for the experiment there are 177 different verbs, so the ten verbs that we have studied represent the 5,6% of the verbs of the sample. This should be considered when we are talking about the effect of the valency information in the syntactic disambiguation process.

Due to the low coverage, they were tested on a corpus containing sentences including the annotated verbs rather than running text.

Wrong applications of rules are mainly due to the occurrence of several verbs with different valencies in one sentence and scope mistakes of the rules. Another reason is low coverage of semantically annotated nouns. The rules involved in the disambiguation of the absolutive-ergative syncretism correctly solve the ambiguity in 64.2% of the cases and incorrectly in 28.5% of the cases, 7% of the cases are left yet ambiguous. The errors are due to the following main reasons:

- Concerning the verb auxiliary tag and the strategy for checking the agreement between the case, the verb and the auxiliary, the main problem is caused by the ellipsis of verbs. This phenomenon is observed in coordination structures and comparatives
- Concerning the valency information: we have on the one hand that the rule disambiguates based on the valency information of an unrelated verb, and on the other hand that the semantic information of the nouns is missing. In order to improve the results generalizing and extending the subcategorization information to more verbs, refining the disambiguation rules based on verb subcategorization and finally improving the semantic noun sets to meet the lexical selection restrictions of the verbs will be necessary.

For a thorough evaluation, the resources need to be improved, first a gold standard of the surface oriented syntax and the implementation of the utility for deriving automatically those correct analysis that have been removed by the CG3 rules. When those are available, a thorough evaluation of more verbs and syntactic disambiguation is planned.

5 Conclusions

This paper has described the work for modeling the surface syntactic disambiguation module reusing and establishing the application order of some of the existing modules in the general framework for the surface syntactic processing. We have started out by setting up the basic steps that should be done in order to improve the syntactic disambiguation module in order to achieve a robust disambiguation as a basic step for further syntactic processes of rich morphology languages like Basque. The results show a modest improvement, although they also present interesting lines for further research. We plan to go further with new experiments based on the combination of the different modules. Furthermore, we would like to apply machine learning-based techniques following the way suggested by Bick (2013) to optimize our grammars.

Finally, future work would certainly profit from access to a larger and revised Gold Standard, to investigate the divergencies on the analyses and to establish the general criteria for solving those differences. In fact our Gold Standard is under development and needs an exaustive study from a qualitative point of view in order to clarify the different sources of error observed when testing our grammars. We should analyze which errors are caused by the grammar and which errors are due to the inconsistencies and incorrect tags in the Gold Standard.

Acknowledgments

This research was supported by the Department of Industry of the Basque Government (IT344-10, S PE11UN114), the University of the Basque Country (GIU09/19).

References

Abney S. P. 1997. *Part-of-speech tagging and partial parsing*. S. Young and G. Bloothooft, editors, Corpus-Based Methods in Language and Speech Processing, Kluwer, Dordrecht.

Aduriz, I., Aldezabal, I., Alegria, I., Arriola, J., Díaz de Ilarraza, A., Ezeiza, N., Gojenola, K. 2003. Finite State Applications for Basque, Workshop on Finite-State Methods in Natural Language Processing. 11th Conference of the European Chapter of the Association for Computational Linguistics, Budapest, Hungary 2003, pp. 3-11.

Agirre E., Atutxa A., Sarasola K. 2012. Contribution of Complex Lexical Information to Solve Syntactic Ambiguity in Basque. Proceedings of COLING 2012, pp: 97—114.

Aldezabal, I., Alegria, I., Ansa O., Arriola, J., Ezeiza, N. 1999. Designing spelling correctors for inflected languages using lexical transducers, Proceedings of European Chapter of the Association for Computational Linguistics, Bergen, Norway, June 1999, pp. 265-266.

Aldezabal, I., Ansa, O., Arrieta, B., Artola, X., Ezeiza, A., Hernández, G., Lersundi, M. 2001. EDBL: a General Lexical Basis for the Automatic Processing of Basque, Proceedings of the IRCS Workshop on Linguistic Databases. Philadelphia (USA), December 2001, pp. 1-10.

Aldezabal, Izaskun, María Jesús Aranzabe, Arantza Díaz de Illarraza, Ainara Estarrona and Larraitz Uria. 2010. 'Euspropbank: Integrating semantic information in the basque dependency treebank', Computational Linguistics and Intelligent Text Processing pp. 60–73.

Alegria, I., Artola, X., Sarasola, K., Urkia, M. 1996. Automatic morphological analysis of Basque. Literary & Linguistic Computing, 11: 193-203.

Alegria, I., Artola, X., Sarasola, K. 1997. Improving a Robust Morphological Analyser using Lexical Transducers. In Mitkov, R. and Nicolov, N. (eds), Recent Advances in Natural Language Processing. Current Issues in Linguistic Theory (CILT) series, 136. John Benjamins publisher company, pp. 97-110.

Maria Jesus Aranzabe, Jose Maria Arriola and Arantza Díaz de Ilarraza. 2009. Theoretical and Methodological issues of tagging Noun Phrase Structures following Dependency Grammar Formalism. In Artiagoitia, X. and Lakarra J.A. (eds) Gramatika Jaietan. Patxi Goenagaren omenez. Donostia: Gipuzkoako Foru Aldundia-UPV/EHU.

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

Bick, Eckhard. 2013. ML-Tuned Constraint Grammars. In: Proceedings of the 27th Pacific Asia Conference on Language, Information and Computation, pp. 440-449. Taipei: Department of English, National Chengchi University.

Didriksen T. 2010. *Constraint Grammar Manual: 3rd version of the CG formalism variant*. Grammar-Soft Aps, Denmark.

DoleŽalova, J. and PetkeviČ, V. 2007. Shallow parsing of Czech sentence based on correct morphological disambiguation. I: Kosta, P. and Schürcks, L. (eds.), Linguistic Investigations into Formal Descriptionof Slavic Languages. Contributions of the Sixth European Conference (FDSL-6), Peter Lang:Frankfurt am Main, 53-64.

Euskaltzaindia. 1994. *Basque Grammar: First Steps (in Basque)*. Euskaltzaindia.

Karlsson, Fred, Atro Voutilainen, Juha Heikkilä & Arto Anttila. 1995. Constraint grammar: A language-independent system for parsing unrestricted text. In Natural Language Processing, No 4. Berlin and New York: Mouton de Gruyter.

Laka, I. 1996. *A Brief Grammar of Euskara, the Basque Language*, Euskararako Errektoreordetza, EHU.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kubler, S., Marinov, S., and Marsi, E. 2007. Maltparser: A language-independent system for data-driven dependency parsing. Natural Language Engineering, 13(2):95.

Samuelsson C. and Voutilainen A. 1997. Comparing a linguistic and a stochastic tagger. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and the Eighth Conference of the European Chapter of the Association for Computational Linguistics. ACL.

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

Tapanainen P. and Järvinen T. 1997. A non-projective dependency parser. In Proceedings of the Fifth Conference on Applied Natural Language Processing ACL.

Wiechetek Linda, Arriola J.M. 2011. An Experiment of Use and Reuse of Verb Valency in Morphosyntactic Disambiguation and Machine Translation for Euskara and North Sámi In: Constraint Grammar Workshop at NoDaLiDa 2011.