View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Repositorio Institucional de la Universidad de Alicante

Procesamiento del lenguaje Natural, nº 41 (2008), pp 5-12 recibido 7-05-2008; aceptado 16-06-2008

Chunk and Clause Identification for Basque by Filtering and Ranking with Perceptrons

Identificación de cláusulas y chunks para el Euskera, usando Filtrado y Ranking con el Perceptron

Iñaki Alegria
University of the Basque Country
649 pk
20018 Donostia
i.alegria@ehu.es

Bertol Arrieta
University of the Basque Country
649 pk
20018 Donostia
bertol@ehu.es

Xavier Carreras
MIT CSAIL
32 Vassar St.
Cambridge MA
02139 USA
carreras@csail.mit.edu

Arantza Díaz de Ilarraza
University of the Basque Country
649 pk
20018 Donostia
a.diazdeilarraza@ehu.es

Larraitz Uria
University of the Basque Country
649 pk
20018 Donostia
larraitz.uria@ehu.es

Resumen: Este artículo presenta sistemas de identificación de chunks y cláusulas para el euskera, combinando gramáticas basadas en reglas con técnicas de aprendizaje automático. Más concretamente, se utiliza el modelo de Filtrado y Ranking con el Perceptron (Carreras, Màrquez y Castro, 2005): un modelo de aprendizaje que permite identificar estructuras sintácticas parciales en la oración, con resultados óptimos para estas tareas en inglés. Este modelo permite incorporar nuevos atributos, y posibilita así el uso de información de diferentes fuentes. De esta manera, hemos añadido información lingüística en los algoritmos de aprendizaje. Así, los resultados del identificador de chunks han mejorado considerablemente y se ha compensado la influencia del relativamente pequeño corpus de entrenamiento que disponemos para el euskera. En cuanto a la identificación de cláusulas, los primeros resultados no son demasiado buenos, debido probablemente al orden libre del euskera y al pequeño corpus del que disponemos actualmente.
Palabras clave: euskera, análisis parcial, chunking, identificación de cláusulas, aprendizaje automático, aprendizaje discriminatorio, perceptron

Abstract: This paper presents systems for syntactic chunking and clause identification for Basque, combining rule-based grammars with machine-learning techniques. Precisely, we used Filtering-Ranking with Perceptrons (Carreras, Màrquez and Castro, 2005): a learning model that recognizes partial syntactic structures in sentences, obtaining state-of-the-art performance for these tasks in English. This model allows incorporating a rich set of features to represent syntactic phrases, making possible to use information from different sources. We used this property in order to include more linguistic features in the learning model and the results obtained in chunking have been improved greatly. This way, we have made up for the relatively small training data available for Basque to learn a chunking model. In the case of clause identification, our preliminary results are low, which suggest that this is due to the free order of Basque and to the small corpus available.
Keywords: Basque language, shallow parsing, chunking, clause identification, machine learning, discriminative learning, perceptron

1 Background

1.1 Basque syntactic parser: an important step toward the grammar checker

In the last years, several works have been done with the aim of building a grammar checker for the Basque language (Ansa et al., 2004); (Diaz De Ilarraza et al., 2005). With that principal purpose, a Basque shallow syntactic parser was created using finite state technologies, constraint grammar rules (Aduriz and Díaz de Ilarraza, 2003) and Hidden Markov Models

ISSN: 1135-5948
© 2008 Sociedad Española para el procesamiento del Lenguaje Natural

based stochastic rules (Ezeiza et al., 1998). Based on the syntactic information extracted by the mentioned shallow syntactic parser, a set of rules was written in order to detect some types of grammatical errors. This way a first version of the Basque grammar checker was developed.

The mentioned shallow syntactic parser is divided into several modules, each one dealing with a different task. First of all, the text is tokenized an analysed morphologically. After that, a tagger/lemmatiser obtains the lemma and the category corresponding to each word form, and another module disambiguates the proposed tags. Then, a rule-based chunker identifies verb and noun phrases, and, finally, a dependency based syntactic tree is obtained by means of a rule-based module.

This parser only recognizes the sentences which are separated by a full stop. Recently, a set of rules has been developed in order to tag sentence and clause splits. However, it has not been integrated in the parser yet.

On the contrary, the rule-based chunker is integrated in this parser and it contains 560 rules; 479 related to noun phrases and 81 related to verb phrases (Aduriz et al., 2004). In Table 1, we present the results of this chunker.

	precision	recall	f-measure
Np	86.92%	80.68%	83.68%
Vp	84.19%	87.77%	85.94%
Chunks	85.92%	83.46%	84.67%

Table 1: Results of the rule-based chunker

In this context, our main goal was to improve the identification of chunks and clauses, using machine learning techniques and combining them with the already existing rules. This way, we would improve both the parser and the grammar checker, due to the fact that the syntactic information used by the grammar checker would be more reliable.

1.2 EPEC: a manually tagged corpus for Basque

In the last years, a big effort has been done to build a manually tagged corpus for the Basque language. This corpus, named EPEC, wants to be the reference corpus for the automatic processing of the Basque language. EPEC is a corpus of standard written Basque which has been manually tagged at different levels: morphology, surface syntax and phrases first, and at deep syntax level later (Aduriz et al., 2006). Half of this corpus was obtained from the *Statistical Corpus of 20th Century Basque* (www.euskaracorpusa.net). The other half was extracted from *Euskaldunon Egunkaria* (www.egunero.info), the only daily newspaper written entirely in standard Basque.

The corpus was tagged semi-automatically. First, it was treated by MORFEUS (Alegria, Artola and Sarasola, 1996), a robust morphological analyser for Basque. This way, the corpus was morphosyntactically analysed giving to each word-form all the possible analysis. Then, this output was manually disambiguated; that is, the correct morphological and syntactic tag was chosen for each word. A similar technique was used to tag the noun and verb chains as well as the sentences and clauses: a rule-based grammar did the first tagging, and the tags were then corrected manually.

This way, 56,000 words (3,708 sentences) were tagged at morphosyntactic level (an average of 15 words per sentence). Chunks and clauses were only tagged in the first 25,000 words. Logically, this one has been the corpus we used in these experiments. This way, our corpus contains the following linguistic information: lemma, part of speech, subcategory, declension, subordinate clauses marks, chunk and clause start-end marks and syntactic functions. Nowadays 300,000 words are being tagged at deep syntax level.

We divided the 25,000 words corpus in three parts: 60% for training, 20% for developing and %20 for testing. We used the development data to evaluate all the models here presented.

1.3 Chunk and clause identification: state of the art

In the last years, machine learning techniques have been applied to different tasks within the NLP field. With respect to chunk and clause identification, the main idea is to recognize partial syntactic structures in a sentence. Since these structures are not very complex, the application of machine learning techniques in this kind of tasks has succeed. Chunk and clause identification shared tasks designed in CoNLL 2000 and 2001, respectively (Tjong Kim Sang and Buchholz, 2000); (Tjong Kim Sang and Déjean, 2001) and the good results obtained with different machine learning

techniques seem to be a clear evidence of its effectiveness.

A key concept behind syntactic chunking is that the chunks which constitute the sentence can be represented as a sequence of labels along the words of a sentence (see Figure 1).

```
   This     is       an          example.
   B-NP    B-VP     B-NP         I-NP

   Hau      adibide       bat        da.
   B-NP      B-NP        I-NP      B-VP
  (this)  (example)     (an)       (is)
```

Figure 1: BIO representation for chunking

Therefore, chunking may be solved using sequential learning models, which predict the most likely sequence of chunk labels given the input sentence. At the heart of these models, there are classifiers which predict the chunk label for a word, given the surrounding context of that word (including the chunk label of the surrounding words). Under this general paradigm, many different algorithms have been applied to chunking. The best systems use discriminative algorithms such as Support Vector Machines (SVM) (Kudo and Matsumoto, 2001), Winnow (Zhang, Damerau and Johnson, 2002), Conditional Random Fields (Sha and Pereira, 2003) or the Averaged Perceptron (Carreras, Màrquez and Castro, 2005). All these algorithms provide important properties. First, in order to represent the data it is possible to incorporate a great deal of different features from many types of sources. Second, they are very efficient algorithms which scale up to the order of tens of thousands of examples and millions of feature dimensions. Third, there are theories that guarantee a good performance of the learned models on unseen data, even in the presence of very large feature sets.

Chunking is evaluated with precision and recall measures of the recognized chunks. To compare the performance of systems, it is common to use the F1 measure (also called F-measure), which corresponds to a weighted harmonic mean of precision and recall, and is computed as:

F1=2*precision*recall / (precision+recall)

In this paper, we will use the F1 measure to compare the different results.

In English, the best systems for chunk identification obtain accuracies at about 94% in F1.

The success of the mentioned methods motivated further research in machine learning systems for recognizing the clause structure of a sentence, a much more difficult problem due to the recursive nature of these structures (see Figure 2). The best systems to date for clause identification obtain accuracies at about 84% in F1. Both systems use a corpus of 200,000 tokens.

```
((Euria ari zuen arren,) oinez joan ginen.)
((Although it was raining,) we went on foot.)
```

Figure 2: Recursive representation of a sentence

Carreras, Màrquez and Castro (2005) took into account the recursive character of clauses, and they developed a system that treats both non recursive and recursive phrases. They suggest a global learning strategy for the general task of recognizing phrases. They propose a filtering-ranking architecture, using perceptrons, and they achieve good results in most of the relevant NLP problems related to recognizing phrases.

2 Phrase recognition using filtering and ranking with perceptrons

As we have seen, Carreras, Màrquez and Castro (2005) suggested a global learning strategy for the general task of recognizing phrases, taking into account the recursive character of some phrases, as clauses.

The system recognizes structures of phrases in a sentence, and it works in two layers. The filtering layer applies simple classifiers to detect boundaries of phrases in the sentence, producing a set of phrase candidates. The ranking layer applies a second set of classifiers that evaluate the phrase candidates produced in the first layer. The final solution is computed with a dynamic programming algorithm that builds the best structure of phrases for the sentence. Depending on the problem at hand, this algorithm will search for sequential or recursive structures of phrases (Carreras, 2005).

All the classifiers are developed using a variant of the Perceptron algorithm: the Averaged Perceptron (Freund and Schapire, 1999), which is a simple improvement of the traditional Perceptron algorithm that learns an averaged combination of classifiers during training. This algorithm has obtained very good results in NLP (Collins, 2002).

Basically, the algorithm keeps visiting training examples in a number of passes or "epochs" on the training set. At each example the algorithm predicts the best phrase structure, and corrects the classifiers if the prediction was wrong, using a very simple rule. We will see in the experiments that the number of passes (epochs) it is not critical at all.

Carreras, Màrquez and Castro (2005) obtained, to date, the third best results for chunking and the best ones for clause identification with this system.

3 Experimental setup

3.1 The corpus

The mentioned part of the EPEC corpus with around 25,000 tokens was used in this experiment: a very small corpus if we compare it with the 200,000 tokens corpus used in the shared task of CoNLL 2000 and 2001 (Tjong Kim Sang and Buchholz, 2000); (Tjong Kim Sang and Déjean, 2001). The EPEC corpus was transformed then to the CoNLL format in order to use the filtering-ranking architecture.

For chunking, the train and the test data consisted initially of three columns (word, part-of-speech and chunk tag). The chunk tags contain the name of the chunk type: B-NP or I-NP for noun phrase words, and B-VP or I-VP for verb phrase words. B-CHUNK mode tags are for the first word of the chunk, and I-CHUNK mode tags for each other word in the chunk. The O chunk tag is used for those tokens which are not part of any chunk. In Figure 3 is an example of the file format for the sentence "Niregana abiatu zen" ("He/She came to me"):

```
Niregana IOR    B-NP
abiatu   ADI    B-VP
zen      ADL    I-VP
.        PUNT   O
```

Figure 3: CoNLL 2000 file format for learning chunks: word, pos and chunk tags in each line.

We have already mentioned that only noun chains and verb chains were tagged as chunks in the Basque corpus. It has to be taken into account that Basque is an agglutinative language and, therefore, prepositions come attached to the nouns or adjectives; that is, the prepositions of other languages as English or Spanish are expressed in Basque as declension marks. That is the reason why prepositional chains were not explicitly tagged. When we have available the 300,000 word corpus, tagged at deep level, we will be able to detect all types of chunks as in CoNLL 2000.

For clause identification, we used the same corpus as the one used for the chunking task.

In this case, the train and the test data consisted, initially, of four columns separated by spaces (word, part of speech, chunk and clause tag). The clause tag may contain the tag (S*, as a start mark; *S), as an ending mark; *, for neither a start nor an ending mark. These tags may be combined recursively.

In Figure 4, we present a real example of the initial training corpus for the sentence "Ogia egunekoa al den galdetzen du." ("He/She asks whether the bread is daily"). Word by word translation: "Ogia (the bread) egunekoa (daily) al den (whether is) galdetzen du (asks)".

```
Ogia       IZE   B-NP (S(S*
egunekoa   ADJ   B-NP     *
al         PRT   B-VP     *
den        ADT   I-VP     *S)
galdetzen  ADI   B-VP     *
du         ADL   I-VP     *
.          PUNT  O        *S)
```

Figure 4: CoNLL 2001 format for clause identification: word, pos, chunk and clause tags for line

3.2 Baselines

For chunking, the baseline results were obtained by selecting the chunk tag which was most frequently associated with the current part-of-speech tag, as in CoNLL 2000. We achieved 54.10% in F1, while 77.07% was obtained with the English training data in CoNLL 2000.

For clause identification, the baseline results were produced by a system which only put clause brackets around sentences, as in CoNLL 2001. We got 37.24% in F1, while 47.71% was obtained with the English training data in CoNLL 2001.

The difference between our results and the CoNLL ones, using the same baseline, shows the difficulty of our starting point. The small training data available for Basque and the fact that this is an agglutinative and free-order language, may explain this difference.

4 *Chunk identification for Basque*

4.1 Initial experiments using filtering and ranking with perceptrons

The same features that those used in CoNLL 2000 were used in the initial experiments with FR-perceptrons: word, part of speech and chunk information. Table 2 shows the best results for the mentioned corpus, with the basic features and the epoch 10.

	precision	recall	f-measure
np	68.12%	68.07%	68.09%
vp	81.42%	86.51%	83.88%
chunks	72.70%	74.03%	73.36%

Table 2: chunker results using the basic features (word, part of speech and chunk tag)

We noticed that the results do not vary much from epoch 10, and the improvements obtained testing the model with further epochs are minimal. That is why we decided to tune the system using the epoch 10, and to test, at the end, the best system with more epochs. In Figure 5, we show the evolution of the performance of the best chunking system, using from 1 to 30 epochs. Note that the result does not improve more than 0.5 points, from the epoch 10.

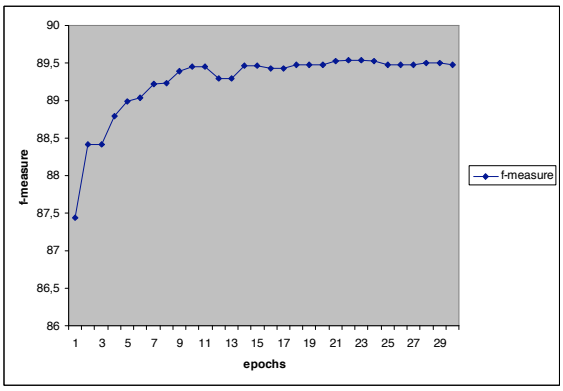

Figure 5: evolution of the performance depending on the number of epochs

4.2 Improvements

We tried to do the stacking with new features extracted from the corpus: lemma, subcategory, declension information, subordinate clause marks and chunking information given by the rule sets. Table 3 shows the results for the chunks, which are a combination of the results of noun phrases and verb phrases:

	precision	recall	f-measure
bf	72.78%	74.21%	73.49%
bf + sc	75.33%	76.61%	75.96%
bf + decl	87.91%	91.05%	**89.45%**
bf + l	74.49%	76.79%	75.62%
bf + soc	73.45%	76.00%	74.70%
bf + sc + decl. + l + soc	85.59%	90.48%	87.97%

Table 3: Basque chunker results, using different type of information (bf: basic features; sc: subcategory inform.; decl: declension inform.; l: lemma; soc: subordinate clause inform.)

As shown in the table before, the best results are obtained using the information of declension. Therefore, we decided to do the final experiment using the basic information plus the declension information and, including, as a new feature, the information which provides the rule-based chunker (see section 1.1). This way we combined machine learning techniques with rule-based grammars and improved the results: 90.16% of f-measure.

4.3 Interpretation of results

The best results obtained in the chunking task are closely related to the target language. The fact that the best results obtained are those in which we added the declension information is a clear evidence of it. In fact, at least one of the words of the noun phrases in Basque has a declension mark. Moreover, the declension mark is easily detected by the morphosyntactic analyser. For example, *big dog* is *zakur handi*, and *with the big dog* would be *zakur handiarekin*. As the lemma of *handiarekin* is *handi*, we know that the word *handiarekin* has a declension mark. For the same reason, we know that *zakur* has not a declension mark. Therefore, *zakur handiarekin* has to be a noun phrase.

It seems clear that delimiting the part of speech and the declension mark facilitates the identification of the chunk. On one hand, the part of speech is important to detect verb phrases. On the other hand, the declension mark is crucial to detect noun phrases, as we can see in the detailed results of noun phrases, when adding the declension information as a new feature:

	precision	recall	f-measure
np	89.60%	92.49%	91.02%
vp	84.55%	88.64%	86.55%
chunks	87.91%	91.05%	89.45%

Table 4: Chunker detailed results using the basic features + declension info

Taking into account that the English corpus is more than 8 times bigger than the Basque one, the results for chunking in Basque are good.

Besides, we have improved the results of the rule-based chunker. However, it has to be taken into account that the test corpus used in both cases is not the same one.

Finally, best results are obtained when stacking the rule-based chunker information to our learning algorithm. This way, we have shown that combining rule based grammars with machine learning techniques improve the results. As a little summary, most important results are compared in Table 5:

technique	features	pr.	rec.	F1
Dependency grammar	-	85.92%	83.46%	84.67%
FR-perceptron	bf	72.78%	74.21%	73.49%
FR-perceptron	bf + decl	87.91%	91.05%	89.45%
FR-perceptron	bf + decl + r.b.ch.info	88.29%	92.11%	**90.16%**

Table 5: Summary of the Basque chunking results (bf: basic features; decl: declension information; r.b.ch.info: rule based chunker information). The corpus used to evaluate the dependency grammar is not the same as the one used to evaluate the FR-perceptron.

We have also analysed the influence of the corpus size to deduce how much the results could increase if we could get a bigger corpus. Best results with the 25%, the 50%, the 75% and the 100% of the initial training corpus are in Table 6:

	Precision	Recall	F-Measure
25%	84.73%	85.56%	85.14%
50%	86.97%	89.78%	88.35%
75%	88.02%	90.95%	89.46%
100%	87.91%	91.05%	89.45%

Table 6: evolution of the performance depending on the size of the training corpus (with basic features and declension information)

Although the results show little improvements, we think that the corpus is too small to draw good conclusions: the training corpus only has 15,000 tokens. Therefore, we are planning to try with a quite bigger corpus.

However, all the results here presented are not fully realistic, since the training corpus was manually tagged. For novel texts, we will have to use the morphosyntactic analyser for Basque, in order to get the necessary linguistic information, which will carry a little decrease in the results.

5 Clause identification for Basque

5.1 Initial experiments using filtering and ranking

The same features that the ones used in CoNLL 2001 were used in our initial experiments in clause identification with FR-perceptrons: word, part of speech, chunk information and clause information. We will call them the basic features.

We trained the filtering-ranking algorithm initially only with the epoch 10 (see Table 7).

	Precision	Recall	F-Measure
clauses	63.67%	41.67%	50.37%

Table 7: Results for basic features

5.2 Improvements

We tried to improve the results stacking the system with new features obtained from the Basque corpus: subcategory, declension information, lemma, information of subordinate clauses and the combination of all the features.

We also did the stacking, adding the information of clause splits, provided by the rule-based grammar (see section 1.1), which improves the results considerably.

Finally, we adapted to Basque a set of features of FR-Perceptron that look for lexical units that trigger clauses. For English, these features look for relative pronouns such as "that, "which", or "who". We created the Basque counterparts for these features, with patterns looking for "non", "zein", “zeinaren”... We call these features "Basque trigger words”. See results in Table 8:

	Prec.	Rec.	F1
bf	63.67%	41.67%	50.37%
bf + sc	63.43%	44.85%	52.55%
bf + d	63.70%	43.87%	51.96%
bf + l	63.18%	45.22%	52.71%
bf + soc	64.13%	44.48%	52.53%
bf + sc + d + l + soc	65.21%	49.39%	56.21%
bf + sc + d + l + soc + cl	67.47%	51.35%	58.32%
bf + sc + d + l + soc + cl + b	69.43%	51.23%	**58.96%**

Table 8: Stacking clause identification system (bf: basic features; sc: subcategory info; d: declension info; l: lemma; soc: subordinate clauses info; cl: rule-based clause identification system’s info; b: Basque triggers)

5.3 Interpretation of the results

It seems that the small corpus we have for the Basque language is the main cause of the low results in comparison with the English ones

(see Table 9). Besides, our preliminary experiments suggest that the clause structure for Basque is very difficult to recognize with partial parsing methods. It has to be pointed out that Basque is a free order language, and therefore sentences may be structured in many different types. The recursive character of clauses does not either facilitate this task.

	Prec.	Rec.	F1
English clause identification	87.99%	81.01%	84.36%
Basque clause identification	69.43%	51.23%	58.96%

Table 9: Comparing Basque and English results on clause identification task

The linguistic features added one by one (subcategory, lemma, declension mark, subordinate clause mark) do not improve so much the results. However, when adding them all together, we get an improvement of 6 points with regard to the results obtained using the basic features. Our hypothesis that subordinate clause marks would improve notably the results has not been completely correct: we obtain the same improvement, adding, for instance, subcategory information. It seems that the more linguistic information we add, the better results we obtain. In this sense, we plan to add information of dependencies, once the Basque automatic parser gives this information.

On the other hand, an improvement of two points is achieved when adding the information of the rules-based grammar developed in order to detect clause splits. This is not either an essential improvement, but it is another little step forward.

But as mentioned, the results are quite low, if we compare them with the English ones, and the one of the reasons seems to be the size of the corpus. That is why we have analysed its influence, measuring the difference between the results obtained with the entire training corpus and the ones obtained using different proportions of the initial corpus. We wanted to deduce how much the results could increase if we could try with a bigger corpus.

Our corpus might be too small even to extract any important conclusion, but it seems that there is quite margin to improve results, increasing its size. In fact, there is a 2 points improvement between using the 50% of the training corpus and using the 100%: a quite big improvement after adding only about 7500 tokens. See Table 10 for more details.

	Precision	Recall	F-Measure
25%	67.94%	48.04%	56.28%
50%	69.31%	48.16%	56.83%
75%	67.99%	50.24%	57.79%
100%	69.43%	51.22%	58.96%

Table 10: influence of the size of the corpus, for clause identification

6 Conclusions and future work

We have used the filtering-ranking architecture with perceptrons for obtaining a competitive chunker and clause identification system for Basque. In spite of using a 8 times smaller corpus than the English one, we have achieved good results for chunking adding new linguistic features. The results for the clause identification system are quite low, although we have improved the initial results stacking the system with linguistic information, derived sometimes from rule-based grammars. Nevertheless, our preliminary experiments suggest that, being the Basque a free order language, this task is more difficult for successful learning, given the available resources.

We also have shown that both in chunk and clause identification, results are improved combining rule based grammars with machine learning techniques.

In the future, we plan to use a bigger corpus to improve the results. The 300,000 words corpus is hoped to be tagged in a quite short period of time. Besides, we are going to add more features, once the Basque automatic parser provides more linguistic information.

We also are going to include the chunker here presented in the shallow parser for Basque, and we will do the same with the clause identification system, if we obtain competitive results. As a consequence, we hope that the grammar checker will also be improved. Besides, a good clause identification tool would help us to detect incorrect commas. For that purpose we would have take into account that all commas would have to be removed for the training corpus, when learning clauses.

These experiments were done using information extracted from a manually tagged corpus. In order to get realistic results, we will have to use a corpus where the linguistic information is obtained with the automatic parser for Basque.

Acknowledgments

We would like to thank Edurne Aldasoro for her help when tagging the corpus.

Research partly funded by the Basque Government (Department of Education, University and Research, IT-397-07), the Spanish Ministry of Education and Science (TIN2007-63173) and the ETORTEK-ANHITZ project from the Basque Government (Department of Culture and Industry, IE06-185).

Xavier Carreras was supported by the Catalan Ministry of Innovation, Universities and Enterprise.

Bibliography

Aduriz I., Aranzabe M., Arriola J., Atutxa A., Díaz de Ilarraza A., Ezeiza N., Gojenola K., Oronoz M., Soroa A., Urizar R. 2006. Methodology and steps towards the construction of EPEC: a corpus of written Basque, tagged at morphological and syntactic levels for the automatic processing. *Corpus Linguistics around the World.Book series: Language and Computers. Vol 56 (1-15). Ed.Wilson, Rayson and Archer. Netherlands.*

Aduriz I., Aranzabe M., Arriola J., Díaz de Ilarraza A., Gojenola K., Oronoz M., Uria L. 2004. A Cascaded Syntactic Analyser for Basque. *Computational Linguistics and Intelligent Text Processing.Pgs.124-135. LNCS Series. Springer Verlag. Berlin.*

Aduriz I., Díaz de Ilarraza A. 2003 Morphosyntactic disambiguation and shallow parsing in Computational Processing of Basque. *Inquiries into the lexicon-syntax relations in Basque. Bernard Oyharçabal (Ed.). University of the Basque Country. Bilbo.*

Alegria I., Artola X., Sarasola K. 1996. Automatic morphological analysis of Basque. *Literary & Linguistic Computing Vol. 11, No. 4, 193-203. Oxford University Press. Oxford. 1996.*

Ansa O., Arregi X., Arrieta B., Ezeiza N., Fernandez I., Garmendia A., Gojenola K., Laskurain B., Martínez E., Oronoz M., Otegi A., Sarasola K., Uria L. 2004. *Integrating NLP Tools for Basque in Text Editors*. Workshop on International Proofing Tools and Language Technologies. University of Patras. Greece.

Carreras X., Màrquez L. and Castro J. 2005. Filtering-Ranking Perceptron Learning for Partial Parsing. *Machine Learning Journal, Special Issue on Learning in Speech and Language Technologies, Vol. 60, Issue 1-3, pgs. 41-71.*

Carreras X. 2005. Learning and Inference in Phrase Recognition: A Filtering-Ranking Architecture using Perceptron. PhD. Polytechnic University of Catalunya.

Collins M., 2002 Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. *EMNLP 2002.*

Díaz de Ilarraza A., Gojenola K., Oronoz M. 2005. *Design and Development of a System for the Detection of Agreement Errors in Basque. CICLing-2005. Mexico.*

Ezeiza N., Aduriz I., Alegria I., Arriola J.M., Urizar R. 1998. Combining Stochastic and Rule-Based Methods for Disambiguation in Agglutinative Languages. *COLING-ACL'98. Vol.1. Pgs.380-384 Montreal (Canada).*

Freund Y. and Schapire R. E. 1999. Large margin classification using the perceptron algorithm. *Machine Learning:*37(3):277-296

Kudo T. and Matsumoto Y. 2001. Chunking with Support Vector Machines. *Proceeding of NAACL 2001, Pittsburgh, PA, USA.*

Tjong Kim Sang E.F. and Buchholz S. 2000. Introduction to the CoNLL-2000 Shared Task: Chunking. *Proceedings of CoNLL-2000 and LLL-2000. Lisbon. Portugal.*

Tjong Kim Sang E.F. and Déjean H. 2001. Introduction to the CoNLL-2001 Shared Task: Clause Identification. In: *Proceedings of CoNLL-2001, Toulouse, France.*

Sha F. and Pereira F. 2003. Shallow parsing with conditional random fields. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*

Zhang, T., Damerau F. and Johnson D. 2002. Text Chunking based on a Generalization of Winnow. In *Journal of Machine Learning Research, vol.2. Pgs. 615-637.*