The use of NLP tools for Basque in a multiple user CALL environment and its feedback

Aldabe I., Arrieta B., Díaz de Ilarraza A., Maritxalar M., Niebla I., Oronoz M., and Uria L.
Department of Computer Languages and Systems
University of the Basque Country
P.O. box 649, E-20080 Donostia

Abstract In this article, we present a Computer Assisted Language Learning (CALL) environment for Basque. The environment has different aims: on the one hand, to offer the users (teachers, learners and computational linguists) different tools and language resources to clarify the linguistic doubts they might have about the language, and on the other hand, to store information about language learners, deviations and errors as the basis for further studies in CALL and Natural Language Processing (NLP). The environment is composed of a workbench (Lentillak), two web applications (Erreus and Irakazi), several NLP tools and two databases (*Errors* and *Deviations*), and it takes in learner corpora as well as native corpora. In addition, we present the experiment we have carried out to evaluate the usefulness of the NLP tools.

1 Introduction

In the last years, there has been a growing interest in the NLP community in CALL, and vice versa. In fact, many new tools, applications and facilities are being constantly marketed. And it is true that, despite there are still some limitations and difficulties when applying NLP tools in CALL, interesting and significant research is being carried out within this field. There are actually many projects, systems and research related to NLP in CALL; to mention some: Loiseau et al. (2005), L'haire and Vandeventer (2003), Nerbonne (2003), Heift (2003), Granger (2003), Granger (2004), etc.

As Nerbonne well points out, the central role for CALL is, or at least should be, to provide comprehensible and understandable materials for both teachers and learners. Indeed, NLP tools can be additional help resources within CALL software in order to enable language learners and teachers to easily obtain information about the target language as well as to get some content materials. In fact, these tools can illustrate linguistic structures, make language comprehensible, provide varied exercise material, spot and correct errors, etc.

In this article, we present a CALL environment for Basque whose aims are to offer the users (teachers, learners and computational linguists) different tools and language resources in order to clarify the linguistic doubts they might have about the language as well as to store information about language learners, deviations and errors[1] as the basis for further studies in CALL and NLP.

[1] We distinguish errors from deviations. We consider an error any ungrammatical output, and a deviation is, for

When integrating our tools in this CALL environment, we took into account that *the use of NLP tools within a CALL software must be designed with care. Giving access to NLP tools is not enough, especially as the target user population is not already familiar with them. Therefore, careful integration of the NLP tools into the didactic concept of the CALL software is a prerequisite to benefit plainly from this innovative technology* (Linguistik online 17, 5/03).

Although Basque is a minority language, some robust NLP tools have been developed for the automatic treatment of the language in the last twenty years. In the IXA research group[2], we have created NLP tools and resources such as a morphosyntactic analyser, a shallow syntactic analyser, monolingual and multilingual dictionaries, a lexical database, a WordNet for Basque and some machine translation prototypes from English and Spanish into Basque. Some of these tools and resources have been now integrated in our CALL environment, and in addition, we have developed new ones specifically for this environment.

In sum, the CALL environment we present here is composed of Irakazi (Aldabe et al., 2005), a teacher-oriented web application for analysing students' information, performances and progress; Erreus, another web application designed for storing information for errors' automatic treatment; two databases (*Errors* and *Deviations*) to store error and deviant instances with different purposes; and Lentillak, a workbench which offers several language resources and NLP tools to help Basque language learners in their tasks.

In the next section, we present the general architecture of the environment. The third section describes the NLP tools integrated in the environment. Section four describes the functionalities of the Lentillak workbench and section five deals with the experiment we have carried out with language learners in order to evaluate how they use and what they think about the integrated NLP tools. Finally, in section six, we outline some conclusions as well as our future lines.

2 The general architecture of the environment

The design and implementation of this environment involve an interdisciplinary approach claimed and followed in (Maritxalar and Díaz de Ilarraza, 1994), (Knutsson et al., 2002), (Greene et al., 2004), (Kraif et al., 2004), and (Vitanova, 2004). According to this approach, learners, teachers and computational linguists work together in the creation of this environment that will be very useful for them. It is very interesting for learners because they can use several Natural Language Processing tools to clarify their linguistic doubts. And at the same time, they enrich the environment with their data. The NLP tools integrated in this environment are also of great value for teachers to consult any linguistic information as well as to analyse the learning process of their students. In addition, teachers also provide the environment with learners' psycholinguistic information and deviant structures. Finally, computational linguists use the environment as a systematic and general framework to store linguistic and technical data to create new NLP tools for language learning purposes as well as for automatic error treatment.

In order to get the objectives we have mentioned before, computer engineers, computational linguists, psycholinguists and language teachers have been working together on the development of this environment. The specific goals are to develop tools for automatic error treatment and analysis as well as to collect learner corpora and store learners' psycholinguistic information.

us, the ungrammatical or inappropriate instances (such as avoidance or reiteration) made by language learners.

[2] http://ixa.si.ehu.es/Ixa

All this will be essential in order to reach our goals.The environment offers in a systematic way a continuous feedback among the tools by means of the constant interaction among all the elements.

As figure 1 shows, we have implemented a workbench and two web applications: Lentillak, Irakazi and Erreus. In the **Lentillak** workbench some NLP tools have been integrated and they are available to clarify the linguistic doubts arisen when producing the target language. The texts written using this workbench are stored as learner corpora. By means of **Irakazi**, teachers store the information about the deviant structures found in learner corpora. It also offers teachers the chance to analyse the collected data in order to know more about the learning process of their learners and they can also specify the strategies to correct the stored deviant structures and use the integrated NLP tools that can be useful for them to correct learners' deviations. In fact, two databases, *Deviations* and *Errors*, are used to collect different aspects of learners' information. In addition, computational linguists add, by means of the **Erreus** application, the technical information corresponding to each error instance for their automatic treatment. Such information is stored in the *Errors* database.

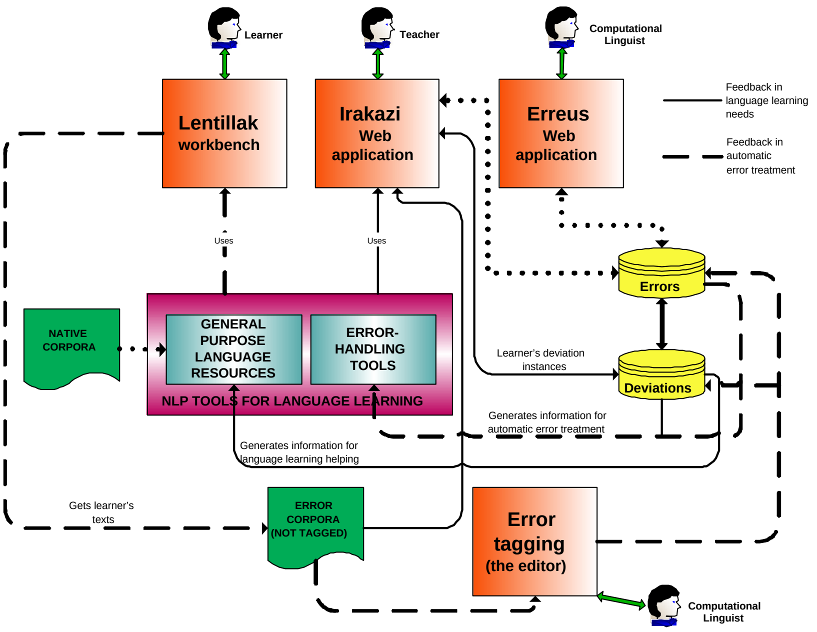

Learner
Teacher
Computational Linguist
Lentillak workbench
Irakazi Web application
Erreus Web application
Feedback in language learning needs
Feedback in automatic error treatment
Uses
Uses
Errors
NATIVE CORPORA
GENERAL PURPOSE LANGUAGE RESOURCES
ERROR-HANDLING TOOLS
NLP TOOLS FOR LANGUAGE LEARNING
Learner's deviation instances
Deviations
Generates information for automatic error treatment
Generates information for language learning helping
Gets learner's texts
ERROR CORPORA (NOT TAGGED)
Error tagging (the editor)
Computational Linguist

Figure 1: The CALL environment.

In the near future, by means of the **error editor tool** (under construction), a computational linguist will manually tag errors and their possible corrections. The results of this tagging process will be used to automatically fill the two mentioned databases.

The **learner corpora** (named error-corpora in figure 1) we collect comprise the free-texts written by learners as well as their exercises. The results of these exercises offer information about the deviations that have been somehow induced by the teacher with pedagogical purposes. On the contrary, the free-texts provide us with the deviant structures students make unconsciously. Besides, it is important to mention that the **native** and learner corpora we make use of are organised according to the language levels set by the Basque language academies.

The **feedback** process in this environment can be defined in such a way that the deviant examples and the information stored in the *Deviations* and *Errors* databases are used to create new rules for automatic error treatment as well as to develop different language resources to respond to learners' and teachers' purposes. "General purpose language resources" and "Error-handling tools", are used in Lentillak and Irakazi, which provide us with learner corpora. And it is important to underline that the information to be introduced in the *Deviations* and *Errors* databases can be obtained by means of the error editor tool or it can be introduced directly through Irakazi.

3 Developing and adapting NLP tools for language learning

Taking into account the needs detected and specified by Basque learners and teachers, we have developed and adapted some NLP tools. We want to remark again the robustness of the tools (morphological analyser, chunker, deep syntax analyser, . . .) used as the basis for the tools we described in this section.

We have grouped the integrated tools depending on whether they are for error detection or not. Therefore, in section 3.1 we describe the language resources concerning error-free data (general purpose language resources), and in section 3.2, we explain the work carried out in the field of automatic error detection and correction.

Basque is an agglutinative language where the constituents of the sentence are freely ordered. Because of this characteristic, linguistic tools such as a declension tool, a conjugation tool, etc. can be indeed very useful.

3.1 General purpose language resources

The adapted NLP tools we present below are based on error-free data and they have been integrated with pedagogical purposes.

- A **morphological information consulting tool** to consult the lemma entries stored in EDBL (a lexical database for Basque with approximately 85.000 entries)(Aldezabal et al., 2001). In this database learners can view examples about how to use the lemmas and obtain morphological information about the entries, improving, this way, their lexical and morphological competence. The student enters a word, the tool lemmatizes it (Aduriz et al., 1992) and then consults it in the database. For example, suppose that the student wants to know if the word "dirudun" (*rich* or *who has money*) is an adjective or not. It may happen that the student does not know the lemma ("diru" or "dirudun") for consulting the word. This is not necessary as the tool obtains the lemma and it makes the query.

- A **conjugation tool** to consult verb conjugations. This tool makes use of the information stored in EDBL and it provides the verb declensions for all Basque verb forms. The users of this tool have the possibility i) to enter a person and number for each agreement marker and the mode/tense for the expected verb in order to get the corresponding auxiliary verb, or ii) to enter a person and number for each of the agreement markers, mode/tense and a root in order to get the corresponding synthetic verb form. This way, learners can get all the possible conjugations of a verb and enrich their knowledge about Basque verb system. For example, if she/he enters the pronouns "hura" (agreement marker=ABS, person=3, number=s) and "guri" (agreement marker=DAT, person=1, number=pl), and the verb root "etorri" (*to come*), the student will obtain the synthetic verb form "datorkigu" (*he comes to us*).

- A **sentence level structure helper**. This tool also makes use of the information stored in the mentioned lexical database. It is very useful for learners to get information about linguistic structures since it provides them with examples about how to use some grammatical structures. This is an interesting tool for learners to familiarise with the linguistic structures of the target language as well as to enrich their grammatical competence. For example, if the student wants to know how to create "comparative" sentences, the tool will provide her/him with the suffix "-ago" (*more*) and examples of its use.

- A **declension generator tool** to find any declined form of a given word. The users have different choices such as i) to enter a word specifying its category (noun or adjective) in order to obtain all its possible declined forms, ii) to choose a word, its category and a declension case in order to get its possible variants (singular definite, plural definite and indefinite forms) of the given word, and iii) to specify a word, its category, a declension case and a declension form (singular definite, plural definite or indefinite) in order to get its declined form. This tool makes use of the morphological generator previously developed in the IXA group. For example, let us think that the student wants to know the inflection of the noun "itsaso" (*sea*) and its dative case in singular (choice iii). Introducing these data the user will obtain the word "itsasoari" (*to the sea*).

- A **KWIC (Key Word In Context) system** that provides access to authentic language use of different language levels. This system searches a word or a lemma in a corpus that has been clustered taking into account the different language levels specified at Basque language schools. The examples obtained from the corpus are displayed in KWIC form and they are a good training corpus for self-study.

- **Hiztegixa**, a web application where learners can look up for a word in several dictionaries (monolingual and bilingual ones) as well as in a corpus, using the same interface.

At the moment, all the explained tools are available in the Lentillak workbench and their usefulness has been already evaluated (section 5).

3.2 Error-handling tools

Below we present the tools we have developed for the detection and correction of both language learners' deviant structures and native speakers' error instances:

- A **spelling checker** which warns users of their spelling errors.

- A **proposal tool** which offers correct proposals when the entered word is wrong. Besides, learners have the option to specify a number of proposals as well as to seek for the most typical error-types. For example, the insertion of the incorrect word "aizpa" will provide us with correct forms "ahizpa" (*sister*) and, "aizka" (*to frighten off*), "zizpa" (*rifle*) . . . This tool is very useful when the students know more or less how to write the word but not its correct spelling. In the future, it will be also possible to ask for proposals specifying learners' language level.

- **Grammar checkers**. For the detection of different error-types, several techniques must be used considering their linguistic requirements. Below, we explain the phenomena we have analysed and the tools we have created based on the collected errors and deviations. The first two techniques are rule-based while the last one makes use of machine learning techniques.

  - **Error detection rules using Constraint Grammar** (Karlsson et al., 1983). We have created some Constraint Grammar based rules to detect errors in some grammatical expressions, determiners and postpositions. Now we are working on a proposal tool at syntactic level and have already got some results concerning postpositions.

  - **Error correction/detection rules applied to dependency trees**. We have designed and developed a system for the detection and correction of *agreement errors* in free texts based on the dependency trees of each sentence, or at least, part of it (Díaz de Ilarraza et al., 2005). The system is composed of three main modules: i) a robust syntactic analyser, ii) a compiler that translates error processing rules, and iii) a module that coordinates the results of the analyser, applying different combinations of the already compiled error rules.

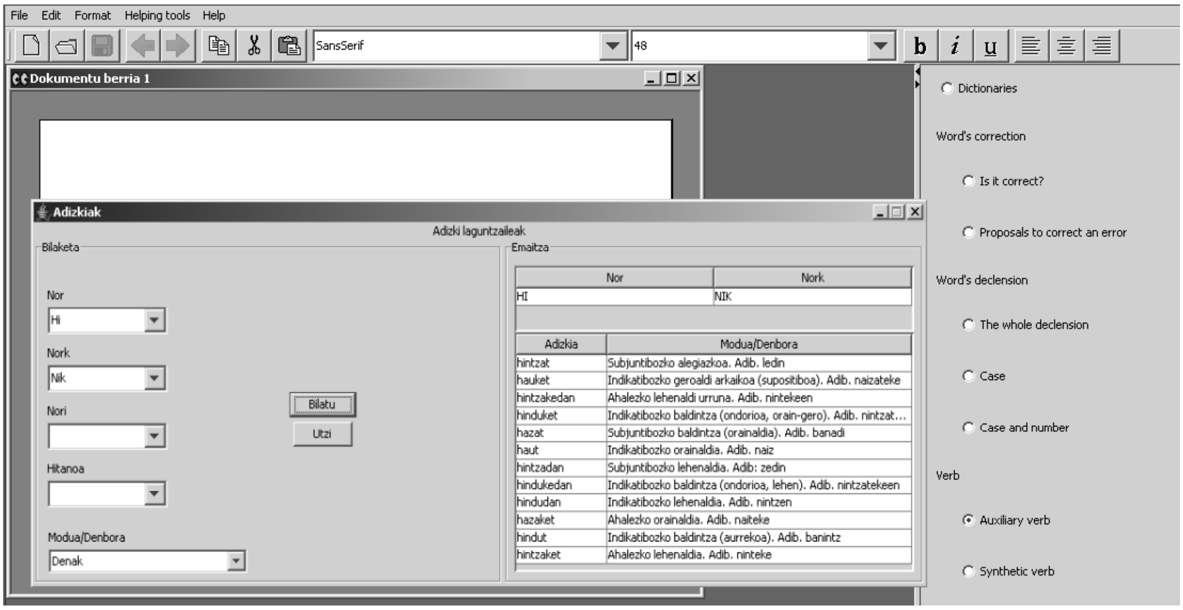
  - **Machine Learning**. Some machine learning techniques have been already applied to detect incorrect uses of the comma. Actually, we have tried to learn commas taking into account that no manual work had to be done, and supposing that commas were placed correctly in the chosen training corpora. In the near future, we will try some other approaches for the same purpose, and we are going to base the learning on clause splitting.

The proposal tool as well as the spelling checker are already integrated and evaluated by means of Lentillak. In the near future, we plan to integrate and evaluate the rest of the grammar checking tools.

4 The functionalities of the Lentillak workbench

Lentillak is the workbench we have used for our experiment in order to evaluate how the NLP tools help learners when making exercises. All the tools mentioned in section 3.1 as well as the spelling checker and the proposal tools described in section 3.2 are available.

The interface of Lentillak (figure 2) offers several functionalities to work with. Depending on their goals, such functionalities have been grouped in different menus. The implemented NLP tools are accessible in the *Helping tools* menu as well as in the right part of the interface. Apart from the specified menus, a notepad where users can take notes of their doubts is also accessible. The environment is implemented in Java, C++ and Perl, and it is based on a client server architecture and the communication with the user is made by means of an intuitive and easy to use web-based interface.



Nor: Hi

Nork: Nik

Nori:

Hitanoa:

Modua/Denbora: Denak

Nor	Nork
HI	NIK

Adizkia	Modua/Denbora
hintzat	Subjuntibozko alegiazkoa. Adib. ledin
hauket	Indikatibozko geroaldi arkaikoa (supositiboa). Adib. naizateke
hintzakedan	Ahalezko lehenaldi urruna. Adib. nintekeen
hinduket	Indikatibozko baldintza (ondorioa, orain-gero). Adib. nintzat...
hazat	Subjuntibozko baldintza (orainaldia). Adib. banadi
haut	Indikatibozko orainaldia. Adib. naiz
hintzadan	Subjuntibozko lehenaldia. Adib: zedin
hindukedan	Indikatibozko baldintza (ondorioa, lehen). Adib. nintzatekeen
hindudan	Indikatibozko lehenaldia. Adib. nintzen
hazaket	Ahalezko orainaldia. Adib. naiteke
hindut	Indikatibozko baldintza (aurrekoa). Adib. banintz
hintzaket	Ahalezko lehenaldia. Adib. ninteke

Figure 2: Lentillak's workbench with the verb conjugation tool displayed.

5 An experiment with language learners

In this experiment, we wanted to evaluate with a questionnaire the usefulness of the NLP tools, i.e. whether the tools are helpful for learners to make exercises. Moreover, our objective was to evaluate if the type of the exercise has influence on the usefulness of the tools (section 5.3). With this purpose, we prepared some exercises to be done by the students using Lentillak.

The experiment was carried out with twenty-five learners of Basque of high level and in three different sessions. Our objective was to measure the usefulness and comprehensibility of each tool integrated in this CALL environment, and, by the way, we also wanted to know if the interface of Lentillak was easy to use and intuitive.

5.1 The exercises and the questionnaire

We designed some exercises to provoke certain linguistic doubts to learners so that they could clarify them using the integrated tools. The type of exercises proposed were: i) try to find the error, ii) make a sentence with a concrete word, and iii) rewrite the sentence changing the verb tense.

In addition to the exercises, we prepared a questionnaire in order to find out i) whether they consider the interface easy to use and ii) whether they think that the NLP tools integrated in Lentillak are useful and easy to use. Respecting to this second point, we specifically made the following questions: a) did you use the tool?; a1) if so: has it been a helpful tool for you to make the exercise?; a2) if not: in spite of not using it, do you think it could be a useful tool?; b) did you understand easily the purpose and the use of the tool?

5.2 Results

As concerns the results of the first point, 88% of the students think the interface is intuitive and easy to use. This high percentage may be, apart from the interface's appropriate design, because the students are nowadays familiarised with this kind of computer based applications.

Table 1 shows the results of the comprehensibility and usefulness of the NLP tools.

	Comprehensibility	Used by	Useful for	Not used but useful for
Dictionaries	96%	88%	86%	100%
Is it correct?	100%	72%	94%	86%
Proposer	75%	20%	80%	80%
Word´s declension	83%	40%	100%	87%
Verb	63%	76%	84%	83%
Sentence level structures	50%	24%	100%	84%
Gradding suffixes	50%	4%	100%	79%
Get examples	71%	16%	100%	71%
Morphological information	75%	12%	67%	68%

Table 1: Comprehensibility and usefulness of the NLP tools.

In respect to the *comprehensibility*, we observe that, almost all the tools are easily comprehensible for more than the 60% of the questioned students. The tools that obtain worse results are the ones to consult sentence level structures and gradding suffixes. We think this could be due to a higher complexity of the grammatical content presented through these tools.

Considering the *usefulness*, the second column shows how many learners have used each tool. The third column displays the percentage of those who, having used the tool, think the tool is useful. And in the last column we see the percentage of the users who, in spite of not having used the tool, think that it could be helpful.

We can also notice that *Dictionaries* (88%), *Verb* (76%) and *Is it correct* (72%) are the most used tools, probably they are very intuitive and because they already exist in environments of common use such as electronic dictionaries and spell checkers.

Most of the students who have used the tools consider that these linguistic resources are really interesting and useful (see column 3), and those who have not used them foresee they can be helpful (see column 4).

We consider very positive that most of the times tools' usefulness is over 80%, even in those cases that learners have not used them to make the proposed exercises.

5.3 Influence of the exercise type for choosing the tools

Students will use some tools or others depending on the type of exercise they have to do. In order to compare the influence of the exercise type in the use of the tools, we additionally asked nine students to write an essay.

As a result of this experiment, we have deduced that the use of the tools is smaller when writing essays. In contrast to the seven tools students made use of to make the exercises, for this task they only used three of them (with different percentages of use): 33% of the students used the *Is it correct?* tool (72% of the students used it when making the exercises); 22% of the students asked for correct proposals of errors (20% in the exercise task) and 11% of them consulted the *Dictionaries* (while 88% used them when making the exercises).

It seems that students consider themselves quite capable for writing an essay without the help of any tool. In other words, if the exercises are not focused on provoking linguistic doubts to students, we foresee that high level learners will rarely consult the tools.

Finally, it is also important to underline that the linguistic phenomena has influence in the usefulness of the tools. Table 1 shows that the tool for consulting *Gradding suffixes* has been hardly used, and this is because it involves very specific language phenomena. In any case, this would involve a deeper research we are not concerned about in this article.

6 Conclusions and Future Work

In this paper, we have presented a CALL environment where several NLP tools have been integrated (some after being adapted and some created precisely for this environment). The environment, which involves an interdisciplinary approach comprising psycholinguistics, computational linguistics and artificial intelligence, is an interesting means to collect learner corpora. Based on these data, we are able to develop new tools as well as to improve the already existing ones. All this continuous feedback is relevant in order to keep enriching the environment.

Within this environment, we find three frameworks: Lentillak, Irakazi and Erreus. The first one is mainly for language learners, the second one for teachers and the last one for computational linguists. And here, we have already integrated wide-coverage and robust tools such as a morphological information consulting tool, a conjugation tool, a sentence level structure helper, a declension generator, a KWIC system, some dictionaries, a spelling checker and a proposal tool at word level. Some more tools will be integrated in the near future. We think that the use of all these tools improve the autonomy of the student in the use of the language.

We have also carried out an experiment with language learners in order to evaluate the usefulness and the comprehensibility of the NLP tools integrated in the environment. For this experiment, we have made use of the Lentillak workbench. The results of the experiment are positive since students have really made use of the implemented tools to fulfil their learning tasks. Besides, they think most of the NLP tools are easy to use and helpful for learning Basque. However, we have corroborated that the use of the tools depends very much on the assigned type of exercises. This experiment has been a first step in the evaluation of the tools, and in a second phase, we will evaluate what learners have learnt by using them.

A Basque language school is already in close cooperation with us to enrich our databases by means of the Irakazi web application. At the same time, thanks to their experience using the NLP tools integrated in this application, we will evaluate how helpful they are also for teachers.

At the moment, using this environment we are developing different CALL systems for automatic essay evaluation and language self-study applications. And finally, we consider that having available some NLP tools for other languages, the environment could become multilingual.

Acknowledgements:

This research is being partially supported by the Basque Government (SAIOTEK project, S-PE04UN11) and the Spanish Ministry of Education and Science (HIZKING21 project, TIN2004-

07918-C04-01). We would like to thank Koldo Gojenola and Aitor Sologaistoa for their help and collaboration in the development of some of the tools we have presented in this paper and to the Jakintza secondary school for their help when carrying out the experiment.

References

Aduriz I., Agirre E., Alegria I., Arregi X., Arriola J., Artola X., Díaz de Ilarraza A., Ezeiza N., Maritxalar M., Sarasola K., and Urkia M. (1992). A morphological analyzer for basque based on two-level morphology. In *Proceedings of the 5th Int. Morphology Meeting*, Krems, Austria.

Aldabe I., Amoros L., Arrieta B., Díaz de Ilarraza A., Maritxalar M., Oronoz M., and Uria L. (2005). Irakazi: a web-based system to assess the learning process of basque language learners. In *Proceedings of a one-day conference Natural Language Processing in Computer-Assisted Language Learning*.

Aldezabal I., Ansa O., Arrieta B., Artola X., Ezeiza A., Hernández G. and Lersundi M. (2001). EDBL: a general lexical basis for the automatic processing of basque. In *IRCS Workshop on linguistic databases*, Philadelphia, USA.

Díaz de Ilarraza A., Gojenola K., and Oronoz M. (2005). Design and development of a system for the detection of agreement errors in basque. In *CICLing-2005, Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, Mexico.

Díaz de Ilarraza A., Maritxalar A., Maritxalar M. and Oronoz M. (1999). Idazkide: an intelligent call environment for second language acquisition. In *Proceedings of a one-day conference Natural Language Processing in Computer-Assisted Language Learning*, p. 12–19: ReCALL.

Granger, S. (2003). Error-tagged learner corpora and CALL: a promising synergy. In *CALICO (special issue on Error Analysis and Error Correction in Computer-Assisted Language Learning) 20(3), pp. 465-480*.

Granger, S. (2004). Computer learner corpus research: current status and future prospects. In *Connor U. and Upton T. (eds.) Applied Corpus Linguistics: A Multidimensional Perspective. Amsterdam & Atlanta: Rodopi. 123-145*.

Greene C., Keogh K., Koller T., Wagner J., Ward M. and Van Genabith J. (2004). Using NLP technology in CALL. In *Proceedings of InSTIL/ICALL2004*.

Heift, T. (2003). Multiple Learner Errors and Meaningful Feedback: A Challenge for ICALL Systems. In *CALICO, 20 (3), pp. 533-549*.

Karlsson F., Voutilanen A., Heikkila J. and Anttila A. (1983). *Constraint Grammar: Language-independent System for Parsing Unrestricted Text*. New York: Mouton de Gruyter, berlin.

Knutsson, O., Cerratto Pargman, T. and Severinson Eklundh, K. (2002). Computer support for second language learners' free text production - initial studies. In *European Journal of Open and Distance Learning (EURODL)*: Martin Valcke and Anne Bruce.

Kraif O., Antoniadis G., Echinard S., Loiseau M., Lebarbé T. and Ponton C. (2004). NLP tools for CALL: the simpler, the better. In *Proceedings of InSTIL/ICALL2004 Symposium, NLP and Speech Technologies in Advanced Language Learning Systems*.

Loiseau M., Antoniadis G., and Ponton C. (2005). *Third International Conference on Multimedia and Information & Communication Technologies in Education (MICTE2005)*. Badajoz, Spain.

L'haire S. and Vandeventer A. (2003). Using NLP tools in a CALL software: the FreeText project.

Maritxalar M. and Díaz de Ilarraza A. (1994). An ICALL system for studying the learning process. In *Computers in Applied Linguistics Conference*.

Nerbonne, J. (2003). Natural Language Processing in Computer-Aided Language Learning. In *The Oxford Handbook of Computational Linguistics, edited by Ruslan Mitkov. Oxford University Press*.

Vitanova I. (2004). Evaluating integrated NLP in foreign language learning: technology meets pedagogy. In *Proceedings of InSTIL/ICALL2004*.