# Knowledge-based intelligent error feedback in a Spanish ICALL system

**Thomas Koller**[1]

**Abstract.** This paper describes the Spanish ICALL system ESPADA which helps language learners to improve their syntactical knowledge. The most important parts of ESPADA for the learner are a Demonstration Module and an Analysis Module. The Demonstration Module provides animated presentation of selected grammatical information. The Analysis Module is able to parse ill-formed sentences and to give adequate feedback on 28 different error types from different levels of language use (syntax, semantics, agreement). It contains a robust chart-based island parser which uses a combination of mal-rules and constraint relaxation to ensure that learner input can be analysed and appropriate error feedback can be generated.

## 1 INTRODUCTION

Many CALL packages fall short when it comes to providing the learner with individualised teaching and flexible feedback [4]. ICALL is very useful in this context as it can automatically generate detailed feedback and provide an individualised environment for each learner. Through this personalisation the learner is more involved in the own language learning process.

The aim of ESPADA is the development and implementation of a fully functional syntax teaching system for German learners of Spanish.[2] In Germany, the teaching of grammar plays an important role in language teaching, but the teaching of syntax in particular is often neglected. Literature pertaining to teaching Spanish syntax in Germany is very sparse. ESPADA aims to provide the learner with a curriculum-independent resource for learning and practising syntactical structures. The project was pedagogically driven and incorporated recent findings on the *real* use of feedback by learners [1]. Currently, ESPADA is directed towards adult learners with basic linguistic knowledge about grammar and syntactic patterns.

The RECALL system (see [4]) is a system that teaches syntax using ICALL resources. It has a Learner Module that contains a model of the individual learner so that the exercises can be tailored specifically to each learner. The Tutoring Module contains an Exercise and a Test Library in order to provide the learner with varied training resources.

ESPADA shares some of the ideas of the RECALL system. However, it is much smaller in scale. For example, it does not have a Learner Module. The Demonstration Module (which is similar to the Tutoring Module of RECALL) of ESPADA provides animated grammars which differ from traditional methods to illustrate grammatical properties. The Analysis Module is able to properly recognise sentences with the wrong type and/or number of sentence components.

The main feature of ESPADA is the multi-faceted preparation and feedback offered to the learner. It consists of three components and their interaction: a Demonstration Module (DM), a Lexicon Module (LM) and an Analysis Module (AM). The Demonstration Module provides animated presentation of selected grammatical information; the Lexicon Module stores and selectively displays lexical information and the Analysis Module dynamically analyses learner input and generates appropriate feedback.

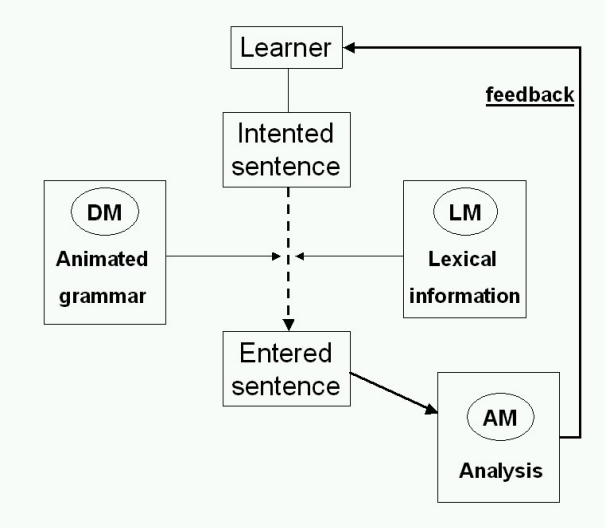


**Figure 1.** Interaction of Modules

ESPADA allows the learner to enter more than a word or phrase. The learner can enter a simple Spanish sentence, which is then analysed and flexible feedback is generated. This feedback can be useful and is usually lacking in commercial language learning systems. The learner can gather further information about syntactic and semantic properties with the help of the Lexicon Module and the Demonstration Module. All the modules can be accessed at the same time.

In Figure 2, the Lexicon Module displays the sentence patterns and verbal forms of *beber* (*to drink*), the Demonstration Module explains the characteristics of nominal groups and the Analysis Module displays the result of analysing the (correct) sentence *El padre bebió un café.* (*The father drank coffee.*)

[1] School of Computing, Dublin City University, Dublin, Ireland, email: tkoller@computing.dcu.ie

[2] Therefore the basic interface language is German. The system has now been enhanced to also have English as the medium of instruction.

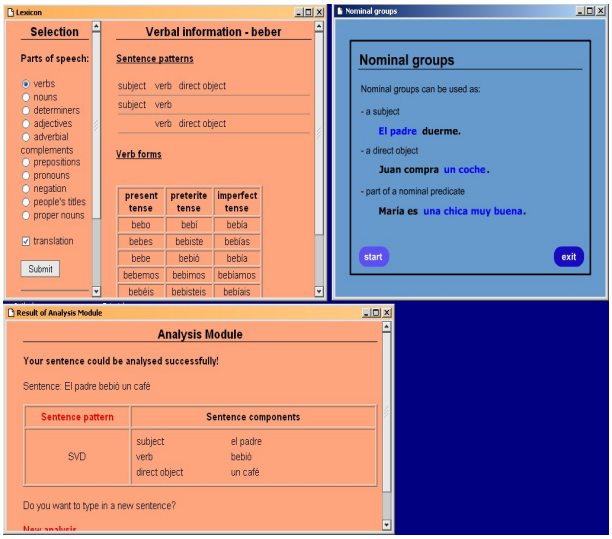


**Figure 2.** Simultaneous use of modules

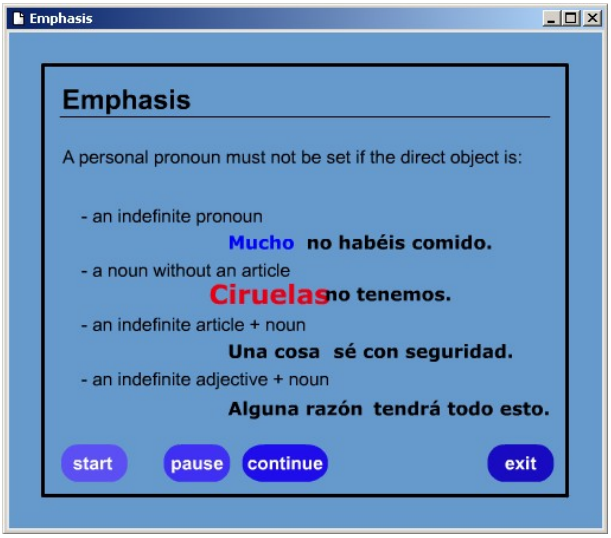


**Figure 3.** Demonstration Module - Explaining topicalisation

ESPADA is based on a detailed knowledge base of linguistic description of syntactic features. It has a modular design with a web-based implementation using HTML and JavaScript. The full form lexicon is coded in XML (based on a XML schema file) and contains 300 lemmas and 1900 full forms. We use a chart-based island parser with mal-rules (see [6], [4]) and constraint relaxation (see [2]).

## 2 DEMONSTRATION MODULE

The Demonstration Module - together with the Analysis Module - represents an animated grammar. The Demonstration Module offers the learner a dynamic representation of grammatical information, which is in contrast with traditional ways of displaying such information. It explains syntactic structures with the help of short Flash animations. These animations can be viewed repeatedly and stopped during rendering, so that the learner has full control of the topics presented (see Figure 3). The Demonstration Module contains 14 different animations, which are classified into 8 main topics (placement of the subject, agreement, types of complements, properties of complements, negations, personal pronouns, *ser* vs. *estar*, topicalisations).

Without the Demonstration Module, the feedback of the Analysis Module would have to be much more extensive. Heift [1] recommends avoiding extensive feedback stating that feedback exceeding three lines was not read by learners.

## 3 LEXICON MODULE

The Lexicon Module (see Figure 4) is interactive and selectively displays the following data in the lexicon:

- all available lemmas of a selected POS
- morphological, syntactic and semantic properties of a chosen verb (allowed sentence patterns, semantic features of required complements and conjugated verb forms)
- different sets of semantic features
- all words of non-verbal POS having a particular set of semantic features

With the information available in the Lexicon Module, the learner can build up a sentence before using the Analysis Module: after choosing an appropriate verb according to the communicative intentions, the learner finds out the required sentence patterns, the semantic type of the complements and the conjugated forms of this verb. Then the learner can see a list of all non-verbal words that correspond to the semantic features required by the verb.

The verb *beber* (*to drink*), for instance, has a possible sentence pattern SVD (subject - verb - direct object) with the required semantic features *+humano* (human) for the subject and *+bebida* (drinkable) for the direct object. Subsequently, the learner can find in the lexicon several personal pronouns and nouns bearing the feature *+humano* and the nouns *café* (*coffee*), *leche* (*milk*) and *vino* (*wine*) for the direct object.

## 4 ANALYSIS MODULE

The Analysis Module is able to perform a detailed analysis of sentences submitted by the learner and to return selective and appropriate feedback to the learner. It can recognize and generate feedback on 28 different error types from different levels of language use (syntax, semantics, agreement). Mistyped words and wrong morphological forms are just handled as errors so that a high degree of robustness of analysis can be ensured. According to [3], very few programs have the ability to give adequate feedback to the learner if an ill-formed sentence is encountered.

The Analysis Module uses a combination of constraint relaxation, different types of mal-rules and a number of pre- and post-parsing tests to ensure that ill-formed input can be analysed and errors are detected.

### 4.1 Range of analysis

The Analysis Module is able to analyse simple Spanish sentences. The range of analysis comprises unmarked sentences as well as sentences in which a nominal phrase or a prepositional phrase appears in

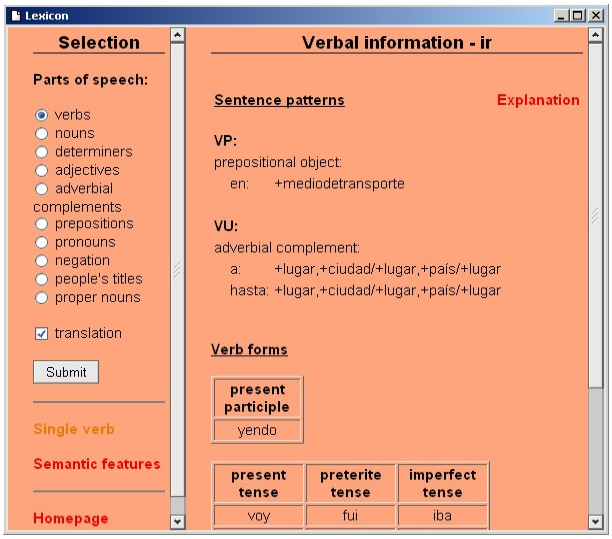


**Figure 4.** Lexicon Module

first position in order to put special emphasis on that part of the statement[3]. The Analysis Module is also able to control the correctness of object pronouns in connection with a topicalized object. It can handle sentences with the simple negation adverb *no*, with complex verbal groups and with up to two adverbial complements.

## 4.2 Main steps of analysis

- Word forms: Syntactic and semantic information of the word forms in the entered sentence are retrieved from the lexical knowledge base.
- Variants: If any word forms of the sentence can belong to several POS, the learner is asked to choose the intended POS for each ambiguous word form.
- POS filtering: With the help of the previously gained POS information, the position of the word forms representing a negation adverb or an adverbial complement are saved. Then these word forms are temporarily taken out of the sentence. This drastically reduces the complexity of the parsing process because different types of adverbial complements can appear at very different places of the sentence. Without the negation adverb or any adverbial complement, any remaining word form of the sentence has to belong to the subject, the verbal group or an object.
- Verbal group: The verbal group is checked through various if-then-tests. It is tested, for instance, if the verbal group starts with a finite verb, if the verbal form of an infinite verb (i. e. past participle, gerund or infinitive) meets the requirements of the preceding verb (e. g. the verb *querer* requires a following verb in infinitive) and if the last verb in the verbal group is a full verb.
- Negation adverbs and object pronouns: The number and position of these POS are checked simultaneously as their positions depend on each other. If any adverbial complements of time were used, their temporal agreement with each other as well as with the finite verb is controlled.
- Determining sentence patterns: If no errors are encountered up to this stage, the basic sentence patterns (e. g. SVD, SVP) of the main verb are extracted from the lexicon. As topicalised sentence patterns (with an object put in first position) are supported, the basic sentence patterns are permutated (here obtaining the sentence patterns DSV, PSV)). Then all these patterns are translated into phrase patterns (e. g. SVD translates into NP|NP and into NP|PP[4]). The sentence patterns and its phrase patterns represent the basic elements for the parsing process.
  If any errors are found, the analysis process is stopped immediately and adequate feedback is given to the learner. The learner can then modify the input and restart the analysis process.
- Parsing: The parsing process is done twice. First, the part of the sentence which precedes the verbal group is parsed against those parts of all phrase patterns which are on the left side of the separator `|`.
  Second, the part of the sentence which follows the verbal group is parsed against those phrase patterns which were parsed successfully in the first parsing phase. It is matched against the part of every phrase pattern on the right side of the separator `|`.
  For instance, given the sentence *El padre dio un libro a su hijo.* (*The father gave a book to his son*) and the phrase pattern `NP|NP;PP`, the phrase *El padre* will be parsed against `NP`. If this parsing process is completed successfully, *un libro a su hijo* will be parsed against `NP;PP`.
- Tests: If any regular sentence patterns of the main verb have been parsed successfully, a broad range of syntactic and semantic checks as well as agreement tests are executed in order to detect corresponding errors. After all the tests have been done, a ranking of sentence patterns is performed according to the number of found errors.
- Displaying results: The sentence pattern(s) with the lowest number of errors are displayed.

## 4.3 Grammar rules

The grammar shows two particular features:

- It has no entries on sentence level (like S → NP VP). The parser only has to analyse single parts of the entered sentence (the parts before and after the verbal group).
- It contains mal-rules in order to recognise faulty nominal phrases representing ungrammatical structures of Spanish (for instance, the mal-rule `NP_Y|Titel;PNn` recognises a nominal phrase which lacks a determiner in connection with a person's title (*señor, señora, señorita*)). As a result, the process does not fail if defective structures of a certain type are encountered. The phrase recognised by the mal-rule will be added to the list of completed phrases (regular nominal phrases are labelled NP_1, NP_2, etc., whereas anticipated faulty NPs are named NP_X and NP_Y). After parsing has been completed, these faulty phrases are identified by a subroutine.

## 4.4 Parser

The parser is a chart-based island parser making use of the Earley algorithm. It only needs to analyse the non-verbal parts of the sentence. The verbal group in simple Spanish sentences has a fixed posi-

[3] Unmarked sentence: *El padre dio un libro a su hijo.* (*The father gave a book to his son.*) Marked sentence: *A su hijo el padre le dio un libro.* (**To his son the father him gave a book.*)

[4] The | sign symbolises the verbal group and is used as a separator by the parser

tion and can only contain certain POS (verbs and prepositions[5]). For that reason it can be examined easily with if-then tests. If an error is encountered in the verbal group, the analysis stops immediately and the user gets an appropriate feedback. This approach helps to avoid as many as error sources as possible before starting the actual parsing process. Using this approach, a broad range of error feedback can be given very fast because the parsing process and the following tests on syntax, semantics and agreement are only due to start if these simple tests have been concluded without any faults.

## 4.5 Mal-rules

The Analysis Module contains two different kinds of mal-rules. These mal-rules recognise faulty structures both at the level of sentence components and within nominal phrases.

The latter type is able to recognize ill-formed input within NPs either concerning wrong word order or missing words. New mal-rules of this type can be easily integrated at a later stage to adapt the grammar to L1-specific learner mistakes or to differing NP structures in other languages.

The type of mal-rules at sentence level is used to parse sentences where the type and/or number of recognised complements does not correspond to the required type and number of complements in sentence patterns of the main verb.

For simple sentences there is a finite amount of possible sentence patterns. This set of sentence patterns is equivalent to all meaningful combinations of the set of complements comprising subject, direct object, indirect object, prepositional object, necessary adverbial complement, for example SV, SVD, SVI, SVP, SVU, SVDI.

After extracting the regular sentence patterns of a given main verb in learner input, all other possible sentence patterns are added to the chart explicitly labelled as mal-rules. Both the lexically induced and the mal-rule sentence patterns are then parsed in the same way. If any regular sentence patterns have been parsed successfully, only these patterns are then checked thoroughly for syntactic and semantic errors as well as errors of agreement. If only mal-rule based sentence patterns have been parsed successfully, the analysis process stops immediately and the recognised sentence components are displayed.

## 4.6 Constraint relaxation

Constraint relaxation (see [5]) represents another important means to properly recognize ill-formed input. In this case it is implemented in terms of successive, layered constraint application and constraint violation bookkeeping. No constraints are applied at all during initial parsing. After parsing has been completed, syntactic and semantic adequacy as well as agreement requirements are controlled via multiple tests. After these tests have been done, the constraint violations are counted for each successfully parsed sentence pattern. The sentence pattern(s) with the lowest number[6] of constraint violations are then displayed.

# 5 GENERATING FEEDBACK

Heift [1] cited the results of a survey about acceptance of feedback as follows:

- Students did not try to correct themselves if no feedback about the type of error was provided.

[5] The only exception to the rule is the conjunction *que* in the expression *tener que*.

[6] This is always a definite number, not a range of numbers.

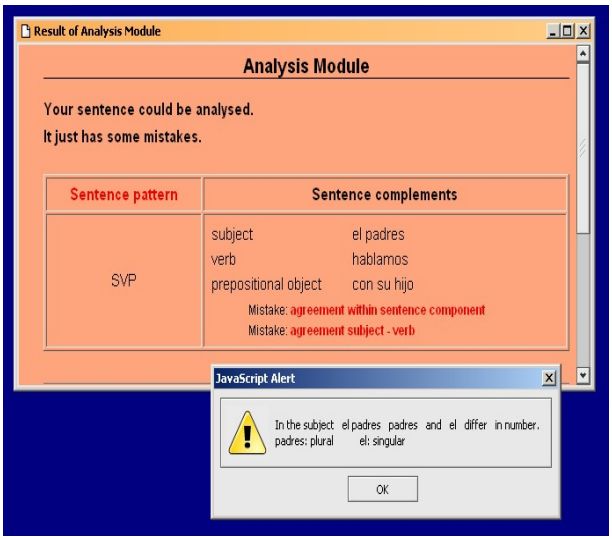


**Figure 5.** Analysis Module

- Feedback exceeding three lines was not read.
- If more than one error was displayed at a time the correction was felt to be too complex.

These results had significant influence of how the feedback in ESPADA was designed. A cascaded feedback system provides two levels of information. If any errors are encountered after parsing has been done, they are all displayed, but only in a shortened form merely giving a hint about the type of each error. Dynamically generated links provide more detailed feedback (which nevertheless does not exceed two lines). This feedback informs the learner of the precise type of error and the values which do not match. Besides that, further general information can be obtained via additional links. (In Figure 5, the browser window displays the result of analysing a sentence with minor errors. The Analysis Module gives information about the recognised parts of the sentence and provides for every error a short explanation combined with a dynamic link. After clicking on the link a JavaScript alert-window opens and gives further information about the error in question.)

There are four different types of feedback after parsing and all constraint checks have been done:

- The sentence could be parsed successfully with at least one regular sentence pattern and no constraint violations were detected: The feedback consists of the successfully parsed sentence pattern(s) and the corresponding sentence components.
- The sentence could be parsed successfully with at least one regular sentence pattern, but constraint violations were detected: The sentence pattern(s) with the lowest number of constraint violations are displayed. The errors are displayed with the help of the cascaded feedback system previously described.
  The errors are not weighted due to the following reason: With the available information the errors could only be organized according to the linguistic type of error (syntactic, semantic, agreement). We doubt that such categories would be of much use for non-linguists.
- The sentence could only be parsed successfully with a mal-rule on sentence level. It contains correctly build phrases, but does not coincide with any regular sentence pattern of the main verb. The

feedback shows the recognised phrases and offers further general information.

- It was not possible to parse the sentence with any sentence pattern (either regular or mal-rule based). The learner gets feedback about this failure of analysis as well as general information about possible reasons.

## 6 CONCLUSIONS & FURTHER WORK

ESPADA offers the learner a comprehensive curriculum-independent means to learn about syntactic structures of simple Spanish sentences. It provides intelligent knowledge-based error analysis and feedback.

A combination of constraint relaxation and mal-rules at different levels has been implemented to ensure that ill-formed sentences can be parsed successfully. Unlike previous approaches using constraint relaxation, this system is implemented in terms of successive, layered constraint application and constraint violation bookkeeping.

The special structural properties of simple Spanish sentences (i. e. easy separable verbal group, fixed position of negation adverb and object pronouns) are exploited in an optimal manner to reduce the input to the parser. The extracted sentence parts are checked on correctness with simple if-then-tests before parsing starts. As a result, the analysis process is speeded up considerably.

ESPADA represents a valuable starting-point for a multilingual ICALL system, which will be developed for several Romance languages (French, Spanish and Italian). Extended versions of the parser and the grammar (containing language-specific rules for several languages) will be very useful to teach contrastive features of Romance languages on the sentence level. Weighted constraints will be deployed to tailor the feedback of the ICALL system more tightly to each individual learner. Priorities can be set either through a database of learner errors or a learner module (see [4]) (saving the previous sessions of a learner).

The feedback of the Analysis Module will be linked directly to corresponding animated grammars in order to increase the learning effect. Animated grammars seem to be rarely used (at least in a systematic manner) and scientific documentation about whether and how to use this kind of grammars is very sparse (see [7]). Therefore the animations developed here may be an interesting starting point for systematic basic research into the design and use of animated grammars.

Currently an online evaluation platform is being created for empirical evaluation of this system. The questions of this platform are tightly linked to the expected features of the PhD work so that the results of this evaluation can be exploited directly for future work.

## ACKNOWLEDGEMENTS

We would like to thank the referees for their comments which helped improve this paper.

## REFERENCES

[1] Trude Heift, 'Error-specific and individualized feedback in a web-based language tutoring system: Do they read it?', *ReCALL*, **13(2)**, 129–142, (2001).

[2] Wolfgang Menzel, *Modellbasierte Fehlerdiagnose in Sprachlehrsystemen*, Niemeyer, Tübingen, 1992.

[3] Wolfgang Menzel and Ingo Schröder, *Constraint-based diagnosis for intelligent language tutoring systems*, Fachbereich Informatik, Universität Hamburg, 1998. Report Nr. FBI-HH-B-208-98.

[4] Maureen Murphy, Anja Krüger, and Andrea Grieszl, *RECALL - Providing an Individualized CALL Environment*, 62–73, Language teaching and language technology, Swets and Zeitlinger, Lisse, 1998.

[5] Veit Reuer, 'Error recognition and parsing of syntactically mildly ill-formed natural language', in *Proceedings of the LFG00 Conference*, pp. 215–225, (2000).

[6] David Schneider and Kathleen F. McCoy, 'Recognizing syntactic errors in the writing of second language learners', in *Proceedings of the 17th International Conference on Computational Linguistics (COLING)*, pp. 1–2, (1998).

[7] Paul A. Sundberg. Animation in CALL: Learning to think in the fourth dimension. Paper Presentation at the CALICO '98 Symposium. online: http://lrs.ed.uiuc.edu/students/p-sundb/homepage/animCALL.html.