Christian Fortmann & Martin Forst

A German LFG for CALL

Donnerstag/Thursday: 12:00

CALL faces specific didactic and technical demands.

- Grammar acquisition in second language learning does not proceed spontaneously, but is a process of conscious rule learning.
- The learner has a native grammar, which may be more or less different from German.
- CALL resources should be designed in a learner-oriented fashion. Communication with and correction by a competent speaker of German get less important.

Interference with the native grammar gives rise to mistakes due to incorrect generalizations. Therefore, a CALL program must be able not only to detect mistakes, but also to specify the incorrect generalization from which they emerge. The learner must be provided with a detailed evaluation of her input and given a strategy to revise the erroneous generalization. Finally, the program should be able to analyze and evaluate any input, not only a set of proposed solutions.

Alongside conjugation and declension, word order is notorious for its importance in learning German as a foreign language. Typical errors, mostly due to interferences with the first language grammar, are:

- more than one constituent in the Vorfeld (prefield): *Peter heute den Kuchen hat gegessen
- more than one verbal element in the V2 position: *Peter hat gegessen heute den Kuchen
- German as an SVO language: *heute hat Peter gegessen den Kuchen

Furthermore, there is a pragmatically and stylistically induced variation in the order of grammatical functions, which means that the acceptability of a word order pattern is to be judged with respect to a particular context rather than in absolute terms. A CALL program should be able to differentiate between marked and ungrammatical word orders.

These demands can be met by a modified LFG for German. In LFG (at least) two syntactic representations are associated with an utterance: c(onstituent)-structure encodes the linear order and grouping of constituents; f(unctional)-structure encodes grammatical functions and features. The framework is well-suited for the implementation of large-scale grammars. The IMS has developed such a grammar, which covers all frequent constructions of German.

The modifications we have made to this grammar provide analyses for utterances that are ungrammatical due to the erroneous generalizations mentioned above, marking them accordingly by a special f-structure feature. The information provided by this feature can then be used to give feedback to the learner and plan the subsequent learning steps.

Marked word orders, which might be incompatible in the context given by a specific task, are identified as such by the attribution of discourse functions to Vorfeld constituents and linear precedence constraints. Mistakes in conjugation and declension can be detected as feature clashes in the f-structure.

With an adapted LFG, it is thus possible to handle the central issues of German grammar in foreign language instruction.

Peter Franke

Pedagogical agents in the Virtual Linguistics Campus: Design and applications

Donnerstag/Thursday: 9:00

State-of-the-art computer-based learning environments built on Internet and multimedia technologies, such as the Virtual Linguistics Campus (VLC), offer rich possibilities for flexible, interactive, and cooperative learning. However, so far, these environments fail to provide adaptive, qualified and motivating support for the individual user in their learning process. “Pedagogical agents” are autonomous software agents, often portrayed as a lifelike character, that can provide this kind of support for human learning by interacting with students in the context of interactive learning environments in order to assist and collaborate with them as appropriate. This talk will present fundamental considerations for the design and application of pedagogical agents in the VLC and other Web-based multimedia learning environments.

Given an appropriate design, pedagogical agents are equally suitable for teaching abstract subject matters (e.g. foreign languages, history, physics) as for helping students learn to perform physical, procedural tasks, such as operating and maintaining complex equipment. An effective pedagogical agent has the following properties: completeness, a balanced design, autonomy, adaptability, expertise, believability, consistency, coherence, trustworthiness, the ability to perceive and change its environment, and the capacity to communicate with other (human and artificial) agents.

The design process that implements these features needs to consider technical, didactic and social aspects. The first group of aspects concerns the programming of the agent and its integration into a Web-based client/server environment (such as the VLC). As a general guideline, agent development for Web-based learning should not rely on proprietary browser plug-ins, but only use established Web technologies (HTML, JavaScript, PHP, etc.) in order to avoid installation problems and facilitate interaction with other components of the learning environment. Didactic aspects of agent design comprise the implementation of effective strategies for teaching and learning, and particularly the development of a sophisticated student modeling component. Didactic design should be guided by the educational paradigm (e.g. instructivist vs. constructivist) chosen for the agent and the learning environment. Finally, building and maintaining strong interpersonal relationships between agents and learners are important social aspects in the design of pedagogical agents because they play a key role in promoting learner motivation. In general, a pedagogical agent requires social intelligence to be able to assess the cognitive and emotional state of the learner correctly and to react appropriately.

Pedagogical agents can play many different roles in interactive learning environments. The following roles are currently being investigated in the context of the VLC: tutor, guide, and librarian. The tutor offers guidance in linguistic problem-solving tasks, and is able to customize support to the learner’s knowledge and skills. The guide explains the structure, navigation, and contents of the VLC to (novice) users, and is available to answer general questions about studying on the VLC. The librarian finds information on the VLC and on the Internet, and provides relevant documents or dynamically generated answers in response to the learner’s questions.

It is essential that students want to use the agent on a regular basis of their own free will because they realize and appreciate its potential to make learning more effective and enjoyable. This can only be achieved by implementing the features listed above in a design process that takes into account technical, didactic and social aspects of pedagogical agents.

Jürgen Handke

Multimedia, computational linguistics and information retrieval in the Virtual Linguistics Campus (VLC)

Donnerstag/Thursday: 10:00

Until recently, computer-assisted teaching was by and large based on offline-systems. These exhibited a high degree of professionality in terms of conception, design, and implementation. In web-based training tools, however, this professional experience seems to have been ignored. As a consequence, most online systems, now generally subsumed under the heading of E-Learning, are restricted to the use of more or less complex text documents with supplementary discussion in class or via specific communicative tools. Advanced principles of tution and the realization of even the most basic elements of multimedia play – if at all – a subordinate role.
Unlike most E-Learning systems, the VLC has acknowledged the importance of multimedia for the presentation of content and the integration of specific tutoring tools leading to several degrees of interactivity beyond simple mouse clicks. Interactive animation, sound, and video, specific graphics for linguistics and the modularization of text are of utmost importance on the VLC. These elements of multimedia are supplemented with tutorial modules that allow the integration of tutorial principles, such as adaptivity and user-modeling.
This paper introduces the main organizational principles of the VLC and suggests ways to integrate methods and priciples leading to more advanced E-learning sites.

Anke Holler & Petra Schulz

How much (computational) linguistics do school children and their teachers need? – ICALL in teaching grammar in elementary school

Donnerstag/Thursday: 11:30

Research on CALL has to date concentrated on computer-assisted foreign language learning (Nerbonne 2003). In this talk we will discuss a further important area of application for (intelligent) CALL systems that has been neglected so far: grammar acquisition in primary school.
Grammar lessons obligatorily start in grade 1. With regard to the crucial aspect of sentence parts, the respective curriculum for German comprises the following topics. In grade 1 and 2 the notion of a sentence is introduced as a "sequence of words which together build an information unit". In grade 3, students work on the subdivision of a sentence in subject and predicate. In grade 4, more fine-grained subdivisions of the sentence such as complex predicates (separable particle verbs, composite tense forms) are discussed. This early grammar teaching aims at enabling students to experience sentences as flexible structures. Therefore, students are engaged in exercises that focus on manipulating sentence constituents in multiple ways. Importantly, these topics encompass a wide range of morpho-syntactic aspects of grammatical analysis. Aspects of constituent structure are touched on as well as the topological structure of sentences, morphological issues and the question of how to classify parts of speech. In order to introduce elementary school students to grammatical analysis, often synthetic and analytic methods are combined. Children may be required to combine

various constituents to form a grammatically correct sentence or to break up a sentence into its constituents and topicalize one of them. Among the media used frequently are written texts and – following the "Gramatikwerkstatt" approach – word cards and sentence strips. However, these media suffer from several disadvantages. Variation of an exercise is only possible to a limited degree. Second, paper media do not permit complete reversibility of the analytical and synthetic processes involved in the grammatical analysis, even though reversibility is a crucial feature in error detection and correction. Furthermore, immediate and individual feedback is not possible.
Therefore, we propose a computer system for grammar lessons that supports elementary school students and teachers in the presentation and practice of these grammar analytic skills. This system has to fulfill at least the following requirements. It should be able (i) to generate relevant practice sentences, (ii) to allow and analyze input by the users, (iii) to recognize and diagnose errors, and (iv) to suggest appropriate corrections. Such a system calls for the implementation of detailed linguistic knowledge of German. In addition, the system requires a combination of various techniques of natural language processing such as parsing and generating sentences and methods for error handling (Jensen et al. 1993, Schmidt-Wigger 1998, Reuer 2000). By integrating corpora (e.g., texts book, children's books) into the system users can be offered authentic and flexible practice material. A prototype that fulfills the above requirements is in development. This system can be used for elementary school students with German as their first language as well as for students learning German as their second language. The latter could start by determining possible constituents in a German sentence.

Gerard Kempen & Camiel van Breugel

A workbench for visual-interactive grammar instruction at the secondary education level

Freitag/Friday: 11:30

Understanding the grammatical structure of the target language is beneficial to certain categories of language learners. This applies, in particular, to foreign language students who are learning to write in a language with many syntax-sensitive spelling rules, and to adult learners who prefer explicit rules (cf. Kempen, 1999). Moreover, grammar constitutes an important body of knowledge in its own right.
In this talk, we describe and demonstrate a visual-interactive grammar workbench that graphically supports students in composing and transforming sentences. The structure of a sentence under construction is displayed in the form of easily interpretable syntactic trees that the student can assemble, modify and move interactively. The system checks on-line the grammatical well-formedness of the structures being manipulated. The Workbench is based on a new psycholinguistically motivated grammar formalism called Performance Grammar (Kempen & Harbusch, 2002).
The system enables, among other things, the design of exercises that allow students to get familiar with, and to apply, syntactic rules underlying agreement, word order, and hierarchical clause structure. The Workbench can record example syntactic transformations and other manipulations like a movie, and play them back at adjustable speed. Grammatical details that are not yet explained or are not relevant for the learning task at hand, can be suppressed. After having observed an example movie, the student is requested to perform similar linguistic operations using other lexical materials.

We conducted a comparative evaluation experiment with ten first-year university students who used some of the Workbench facilities to refresh their high-school knowledge of grammatical concepts. Ten other students followed the same course doing their exercises with a computational version of traditional, non-visual instruction. The Workbench produced favorable learning results.
The Workbench will be demonstrated as an interactive training tool for learning to apply syntactic rules of Dutch. The system runs as a JAVA applet under modern internet browsers.

Rafiq Khader, Tracy Holloway King & Miriam Butt

Deep CALL grammars: The LFG-OT Experiment

Mittwoch/Wednesday: 16:30

The use of computers for composition and particularly their use in writing classes prompted the call for software agents to help writers edit their work in terms of grammar, style and language. This need is particularly great among non-native writers of a language. Among the many programs created were: the Writer's Workbench (Wong 1996), Ruskin (William 1992), Expert Editor (Johnson 1985), Epistle Program (Wong 1996), FROG (Imlah & du Boulay 1985) LINGER (Yazdani 1991), E-linger (Lawler 1991), Grammatik (Liou 1992, Yu and Davies 1996), Native English (Tschichold 1999) and the Microsoft Word program (Markoff 2002). Though some use underlying NLP-based applications, the full potential of NLP-based grammars as error checkers has yet to be realized (e.g., Holland and Kaplan 1995).
This paper explores and evaluates the potential of using a broad-coverage English LFG-based grammar (Riezler et al. 2002) developed within the Parallel Grammar project for parsing, generation and machine translation (Butt et al. 1999) as a grammar checker. The grammar has been augmented with a version of Optimality Theory (Prince and Smolensky 1993, Frank et al. 2001.) to allow it to handle ungrammatical input and provide feedback as to why the input was ungrammatical. In particular, the output structure for an ungrammatical sentence indicates not only that it is ungrammatical, but also which part of the structure is ungrammatical and why. For example, a sentence like John happy. would have a mark stating missing-copular at the level where the verb should have been. Sentences can have more than one mark if more than one error is found. For example, a sentence like Book red. Would have the missing-copular mark at the level where the verb should have been and a missing-determiner mark at the level of the subject noun phrase. Apart from its capacity to parse, the grammar can be reversed to generate as well (Shemtov 1997). The generator can thus be employed to suggest what the grammatical input should look like based on the ungrammatical and can provide appropriate feedback so that second language learners can understand and hence correct the errors.
One of the greatest criticisms levelled against grammar checkers is their use for the general population, especially ones aimed at both native writers and CALL students. Thus, in order to make the LFG-OT grammar checker more reliable (Tschichold 1999a), a select group of learners was identified: Chinese learners of English as a second language. The list of errors predominant among this group was identified, e.g., missing determiners, missing copulas, lack of uppercasing on proper nouns (Zheng 1993; Chen 1997). The errors were classified as lexical, grammatical, and mechanical. However, for this evaluation, only grammatical and mechanical errors were identified because the NLP technology to identify errors due to subtle lexical semantic effects is not yet advanced enough in the underlying LFG-OT grammar. As recommended by Tschichold (1999), a learner corpus, i.e., a database consisting of

spontaneous or elicited discourse produced by second language learners of English, was collected as a way to specialize the grammar checker for this particular group of learners (Granger 1996). Once the set of errors to be flagged were identified, the adaptation of the existing LFG-OT grammar to a CALL grammar took about a day.
To test and enhance the capacity of this grammar, a list of fifty sentences comprising grammatical and mechanical errors was identified. The LFG-OT grammar checker was able to detect 23 out of the 50 errorsThis performance is far better than the result obtained with the Microsoft Word 2002 post editing tool which only detected 10 errors and provided less helpful correction suggestions to the language.

Lothar Lemnitzer

MiLCA: The use of media in learning linguistics

Mittwoch/Wednesday: 17:30

The aim of this project has been to generate and supply high-quality content in the core fields of a CL curriculum on a national as well as an international level. Learning objects and modules which have been developed at five German universities have been used, first, at the developing site and later on by the partners and interested external sites simultaneously. The didactic design of the courses follows the three learning scenarios of Web-based training, virtual course and (two-semester) project. All partners have used didactic consulting facilities of the Institut for Psychology, University of Tübingen. The same partner is also responsible for a formative and summarizing evaluation of all teaching activities.
The project ended in 2003. All results are available to the interested scientific community. For their distribution, a portal has been established with the co-ordinating project partner (http://milca.sfs.uni-tuebingen.de) and a small enterprise for the management of the digital rights has been founded by three of the partners.
A major achievement of the whole project has been the development (by the Univ. Giessen partner) and consistent use of a Document Type Definition for learning modules which draws on the “Learning Object Metadata” (LOM) standards established by the IEEE. The content model of this document type has been used woth only minor changes in the latest version of th ILIAS Learning Management System.
The learning modules which have been developed so fare comprise: Algorithms for Computational Linguistics, Computational Semantics, Signal processing and spoken language, Grammar Formalisms und Parsing, Text Technology, Information Retrieval, Computational Lexicography, Dialogue Systems, and Language Technology for Intelligent Computer Assisted Language Learning.
Various course modules might be of interest for the teaching in general linguistics as well, e.g. parts of Computational Semantics, Computational Lexicography and I-CALL. It is the explicit wish of the developers that their modules be used in other contexts. The further development and maintenance of the teaching material which follows an “open content” approach is also envisaged.

Arantxa Martín-Lozano & Detmar Meurers

Issues in a grammar-based intelligent tutoring system for learning Spanish grammar

Freitag/Friday: 13:30

In this talk we present our work on Tigre ('Tutor interactivo de gramática del español'), the prototype of an Intelligent Tutoring System (ITS) which incorporates grammar-based NLP technology and provides English-speaking learners of Spanish as a foreign language the opportunity to practice their Spanish grammar skills through cycles of on-line exercises with partially free learner input and intelligent feedback to learner's errors.

A number of CALL systems for practice of Spanish grammar are already available to learners both on the web and in commercial CD-ROMs (cf., e.g., http://www.quia.com/dir/spanish/ and http://www.colby.edu/~bknelson/exercises/). However, most of these systems rely on pattern-matching as their basic technique for processing learners' input and thus neither give the learner the opportunity to use the target language creatively (the correct answers are completely predetermined by the system) nor provide intelligent feedback to errors by analyzing the structures containing them and hypothesizing underlying students' misconceptions.

Tigre overcomes these problems through

a) the use of a state-of-the-art constraint-based processing system (Trale, Meurers et al. 2002) in which a basic Head-Driven Phrase Structure Grammar (HPSG, Pollard and Sag 1994) of Spanish has been implemented and augmented with mal-rules (Weischedel et al. 1978) which facilitate the processing of ungrammatical sentences generated by beginning and intermediate learners of Spanish whose native language is English;

b) the use of an instructional component which provides appropriate feedback to errors depending on the mal-rules used for parsing the sentences generated by the learner. The question how feedback about multiple errors can be integrated has started to receive more attention in the recent literature (Heift 2003) and we will discuss how Tigre collects and evaluates both exercise-specific and general errors in an integrated way and compare it to approaches avoiding mal-rules (cf., eg., Reuer 2003).

The curriculum of Tigre focuses on key grammatical problems for native speakers of English learning Spanish. The full system will cover the properties of Spanish subjects and objects, the behavior of clitic pronouns and the mapping of arguments into grammatical functions for different kinds of Spanish verbs, which often differ significantly from their English counterparts. The exercises presented by the tutor to the learner focus on the use of particular syntactic structures and require semi-free learner input, such as building sentences where some of the words are given by the system and answering to questions. Exercises which require semi-free learner input are of pedagogical and computational interest. From the pedagogical point of view, these exercises succeed in leading the students to the use of the verbs and constructions that are the focus of the given exercise and in helping learners at the beginning and intermediate levels to build vocabulary. From the computational point of view, these exercises help to keep the learners' input in the bounds of the coverage of the grammar implemented in the system.

Veit Reuer

In search of the correct error - yet another approach to parsing ill-formed language

Freitag/Friday: 12:30

Parsing learner language either for precise error-feedback in CALL-applications or as the basis for psycholinguistic learner language analysis has seen some attention in the last few years (e.g. Granger et al. 2001, Heift & Nicholson 2001). However most of these approaches either cover only a limited set of errors or have a reduced grammar coverage with the notable exception of Menzel/Schröder 1998.

We have developed a chart-based parser which utilizes LFG (Bresnan 2001) and recognizes errors anticipation-free. Two types of structures are used to build up a description of a sentence in parallel, namely feature structures and phrase structure rules. So called f-structure feature-clashes are handled to identify errors in agreement, government and other verbal features such as finiteness. A modified unification method ensures that conflicting values are integrated into the f-structure as values of a special error feature in order store the information. Additionally the proposed unification algorithm has the advantage of being only a minimal extension retaining the relevant properties such as monotonicity as well as reflexivity and transitivity.

The second type of errors handled are linearization errors (omissions, deletions, spurious replacement, permutation errors). The method is similar to the 'one-parse-only' approach presented in Lee et al. 1995 with major modifications for limiting the search space and storing the cause of an error. Following the application of the *fundamental rule* in an Earley-style parsing algorithm, error-marked items are added to the chart representing the different error hypotheses, if there is structural evidence for an error in the chart. Furthermore the recognition of certain phrasal permutations is integrated into the 'one-parse' approach.

The parse with the fewest cases of error-incorporations/corrections necessary is chosen as the final description of the sentence. Furthermore the error search is restricted to linguistically well defined areas in terms of features and part of speech based on a statistical analysis of the corpus in Heringer 1995 (7107 sentences).

In order to evaluate the parser, a small corpus with actual learner language has been set up. The corpus is composed of 130 sentences with morphosyntactic errors from learners of German using an intelligent CALL-application from Heringer 1995. These sentences were manually annotated for the most probable morphosyntactic error.

Given that the lexicon covered the corrected versions of the sentences, the system was able to identify the expected error type(s) for 53 % of the erroneous sentences, the expected type among others for 18 % and incorrect errors types for another 19 % of the sentences. 10 % of the sentences could not be analysed either because of the parse exceeding a (arbitrary) limit of 5000 chart items or a too heavily distorted word order. Only few sentences contain multiple errors, since the sentences are rather simple (av. sentence length ~ 7) and no false positives were analysed.

Even though this type of evaluation of a parser specifically analysing learner language in an ICALL-system seems crucial, it has rarely been used to our knowledge (but see e.g. Eagles 1996).The evaluation demonstrates that the parser is able to identify the majority of error types correctly using an unmodified computational grammar and lexicon. Adding weighted error types will probably increase the explanation capabilities of the parser even more. Additionally the grammar theory LFG used here has the advantage of being highly similar to

concepts used in grammar textbooks which allows for the simple and adequate generation of feedback.

Cornelia Tschichold

NLP in CALL: Taking small steps

Mittwoch/Wednesday: 14:00

So far, no truly NLP-supported CALL programs have made it into the commercial market. The reasons for this apparent failure are relatively easy to find. They can be grouped into two sections, the first one to do with the type of input a CALL program has to deal with, the second one covering the output expected of a CALL program. As feedback on their language production is essential to language learners, it is absolutely necessary to evaluate the language produced by the user of a comprehensive CALL program at some point. The program will therefore have to deal with linguistic input and produce feedback for the learner.

NLP techniques have been developed for applications other than learner language, applications that deal with mostly correct, native-speaker language (preferably English), and they typically do not function well enough when used on a different kind of input, i.e. learner language that contains a substantially higher number and a different quality of errors than most texts. From a user perspective, the requirements for CALL programs are fundamentally different from NLP applications such as MT, information retrieval, etc. Whereas such NLP applications are aimed at a native speaker user, and typically have to deal with texts from a very specific domain only, CALL programs normally have to cover general language with its much higher degree of ambiguity. The input to an NLP component in a CALL program is thus considerably different from the typical input of most NLP tools. On the output side, the biggest problem is the fact that the tolerance rate for errors or even for superfluous messages produced by the program is close to zero. In a CALL context it is simply not acceptable to give incorrect feedback to program users because these users have a strong tendency to view the program as the authority on the language they are trying to learn. We clearly cannot expect language learners, especially beginners – the group of learners most CALL programs are aimed at – to use common sense judgements in a foreign language when evaluating the program's responses to the language they want evaluated.

It is thus perfectly understandable why commercial CALL programs keep using the same basic techniques they have been using for decades: MCQs in all forms, fill-in-the-blank exercises, etc. These techniques are reliable, but only work for data already in the program. They are of no use when dealing with free input. To my knowledge no commercial CALL program today tries to use NLP tools to analyse text written by the learner. As long as parsers are not considerably more reliable and as long as we do not have at least a rudimentary semantic analysis that works on non-specialist texts, this is a sensible policy. Still, considerable improvements on today's typical CALL programs are possible, and that some NLP tools could be used in CALL programs today. Some significant improvement is still possible without the help of NLP tools, but only the use of dedicated NLP tools will allow us to supply truly useful feedback to free input learner language. What we have today in terms of reliable technology are large-scale lexicons and morphology analysers for a number of major languages. These could be made use of to improve the feedback given to learners on the level of single words and short phrases. Given that most learners see vocabulary as the biggest obstacle on their way to success, it would make sense for CALL designers to put more emphasis on vocabulary and intelligent methods of giving feedback to the language produced by learners on this level. Assuming that learner language is typically less rich in terms of

vocabulary and grammatical constructions, we could also imagine that parsers could be adapted to this type of language and used to find typical errors. Realistically, however, we should not expect a general parser to work well in a CALL program. In my view, CALL is not a field where general NLP tools can be used to generate easy money; CALL requires tools dedicated to the task at hand.

Heike Zinsmeister

Learning verb object collocations for text production

Mittwoch/Wednesday: 15:00

This talk introduces an example of exploiting a statistical grammar model for language teaching. The grammar model provides ranked lists of verb object pairs which allow to determine typical collocations. This information is highly relevant for language learners, especially in tasks of text production.
The grammar itself learns cooccurrence frequencies of verbs and their objects from unannotated German text. These corpus-based frequency counts often result in uneven distributions: given a specific verb, for example, we typically find a relatively small number of frequently cooccuring objects and a relatively large number of infrequent ones. Among the frequent combinations, there is a high proportion of collocations and other idiomatically fixed combinations which provide a rich knowledge base for language learning.
One task for the learner is to find an appropriate verb for a given nominal object which includes the learning of support verb constructions. The data is presented to the learner by means of an interactive tool which displays German nouns and requires the learner to fill in a missing verb. The input is then compared with the corresponding frequency list and either confirmed or rejected.
Afterwards, a list of highly ranked verbal collocates is presented to the learner. For example, the verbal candidate for the object 'Vorschlag' will be compared with a list starting with the verbs 'machen', 'unterbreiten', 'ablehnen', 'vorlegen', 'erarbeiten', 'prüfen', 'begrüßen', etc., which will then be shown to the learner.
Another task is the learning of typical object choices given a specific verb, such as 'einlegen' and its objects 'Beschwerde', 'Berufung', 'Widerspruch', etc. The system also provides a special information about verbs which is normally not even specified in dictionaries: it informs the learner if a frequently used verb does not show any significant preferences on the side of its object.