Interactive sentence combining and paraphrasing in support of integrated writing and grammar instruction: A new application area for natural language sentence generators

Karin HARBUSCH, Camiel VAN BREUGEL, Ulrich KOCH & Gerard KEMPEN
University of Koblenz-Landau, Computer Science Dept.
Universitätsstraße 1
56070 Koblenz, GERMANY
{harbusch, camiel, koch}@uni-koblenz.de

Max Planck Institute
PO Box 310
6500 AH Nijmegen, NETHERLANDS
gerard.kempen@mpi.nl

Abstract

The potential of sentence generators as engines in Intelligent Computer-Assisted Language Learning and teaching (ICALL) software has hardly been explored. We sketch the prototype of COMPASS, a system that supports integrated writing and grammar curricula for 10 to 14 year old elementary or secondary schoolers. The system enables first- or second-language teachers to design controlled writing exercises, in particular of the "sentence combining" variety. The system includes facilities for error diagnosis and on-line feedback. Syntactic structures built by students or system can be displayed as easily understood phrase-structure or dependency trees, adapted to the student's level of grammatical knowledge. The heart of the system is a specially designed generator capable of lexically guided sentence generation, of generating syntactic paraphrases, and displaying syntactic structures visually.

1 Introduction: sentence combining

In many countries, a satisfactory level of writing proficiency is increasingly being recognized as an important goal of first- and second-language instruction at all levels of education. In response to this trend, language technology is beginning to contribute computational tools for writing curricula—(semi-)automatic essay grading being a recent example (e.g. Shermis & Burstein, 2003). The software system described in the present paper supports elementary or secondary schoolers in developing the SYNTACTIC aspects of their writing skills, with German as target language.

The following little story was written by a 10 year old German student as part of a writing exercise. It comprises 15 short sentences, each consisting of a single finite clause. The first and last sentences of the text[1] are as follows:

(1) *Die Kinder wollen zum Mond fliegen.*
The children want to-the moon fly
(2) *Sie bauen eine Rakete. [...]*
They build a rocket
(3) *Sie fliegen nach Hause.* (4) *Zu Hause*
They fly to home At home
erzählen sie alles ihren Eltern.
tell they all (to) their parents

An important goal of writing instruction in elementary and secondary schools in Germany and elsewhere is to raise the level of syntactic diversity of the texts produced by the students. Combining simple clauses into complex or compound sentences is one of the means to this goal. For example, the author of the present story could have combined sentences (1) and (2) as in (5), or (3) and (4) as in (6).

(5) *Die Kinder bauen eine Rakete, weil sie zum Mond fliegen wollen.* [*weil* 'because']
(6) *Sie fliegen nach Hause und erzählen alles ihren Eltern.* [*und* 'and']

At the end of the 1960s, "sentence combining" originated in the United States as a form of "controlled writing" exercises, and various em-

[1]From sentence material collected by our partners at the Psychology Department of the University of Koblenz-Landau under research grant "Wissen-schaf(f)t Zukunft" from the Ministry of Education, Science, Youth and Culture of Rheinland-Pfalz, Germany. The work presented here is partially funded by that grant.

pirical evaluation studies have since confirmed its usefulness (Daiker et al., 1985). In a sentence-combining exercise, students are presented with a sequence of short clauses each expressing a simple proposition. Together, the propositions make up a little story or essay. By transforming the short clauses and combining them into longer sentences, the students then produce a coherent and fluent piece of text. Exercises are often accompanied by instructions to combine clauses in a particular syntactic way (e.g. "use a relative clause"). This requires understanding by the student of grammatical terminology. Actually, empirical studies show that writing instruction as well as grammar teaching yield better results when trained in an integrated manner than when trained in isolation (e.g. Mellon 1969; Schuurs, 1990).

Currently, computer support for sentence combining is restricted to multiple-choice questions or quizzes. To our knowledge, no software tool currently exists that deploys generation technology to evaluate student responses to sentence combining exercises. As a matter of fact, virtually the entire literature on the application of NLP to the syntactic aspects of first- and second-language teaching is based on syntactic parsing technology (Heift & Schulze 2003). To our knowledge, Zamorano Mansilla (2004) is the only project that applies a sentence generator (KPML; Bateman 1997) to the recognition and diagnosis of writing errors ("fill-in-the-blank" exercises, not sentence combining).

The system introduced in the present paper is a first attempt to fill this gap. It supports students in producing diverse sentence structures on-line while focusing on grammatical structure, i.e. without the need to pay much attention to semantic content generation, word inflection, spelling, and typing. The system evaluates grammatical correctness of student-generated output and compliance with the task assignment online, and provides accurate feedback.

2 The COMPASS system

The kernel of the COMPASS system (for COMbinatorial and Paraphrastic Assembly of Sentence Structure) is a specially designed sentence generator capable of LEXICALLY GUIDED SENTENCE CONSTRUCTION, and of PARAPHRASING. It takes as input (1) a set of lexically anchored "treelets" that specify the subcategorization frames of the lexical anchors, together with (2) a specification of the grammatical relations between lexical anchors that the to-be-generated sentences should realize. (This in contrast with familiar generators that take a semantic structure as input.) The key structure building operation in the generator is DISJUNCTIVE FEATURE UNIFICATION (for details of the underlying Performance Grammar formalism, see Kempen & Harbusch 2002). Moreover, instead of generating a single sentence as output, it produces the full set of well-formed syntactic paraphrases licensed by the current lexical input in conjunction with the grammar (Harbusch et al., 2006).

The user interface lets the student describe a visually displayed scene by selecting words from a list of inflected word forms. To this purpose, s/he drags the word forms out of the list and drops them into the system's workspace on the screen, where COMPASS displays the treelets associated with them (Figure 1). Then, the student combines them in accordance with the required grammatical relations, and orders the branches of the resulting hierarchical structure from left to right (also by drag & drop), until s/he judges that the word form string dominated by the tree (which may contain crossing branches) expresses the intended meaning in the form of a grammatically correct sentence. The generator produces all possible word-order paraphrases and checks whether the terminal string of at least one paraphrase is identical to the string produced by the student. If not, COMPASS attempts to diagnose the error by checking if the latter string could have resulted from mis- or non-application of a linear order rule, and provides feedback accordingly. The rule base of the system also includes MAL-RULES, which generate structures occurring in frequently observable errors. If the generator has to apply a mal-rule in order to match student output exactly, a feedback message is displayed (e.g. "Don't use main clause word order in a subordinate clause").

Notice that, because the students compose all sentences and phrases under generator control, COMPASS can evaluate their responses WITHOUT THE HELP OF A PARSER: Based on its paraphrastic capabilities and its mal-rules, the system

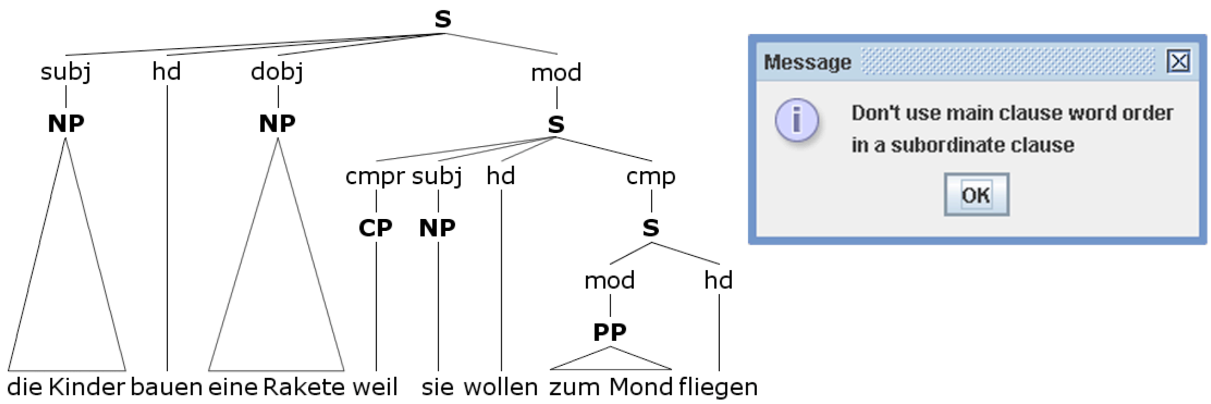

Figure 1. A COMPASS exercise for English speaking students learning German as a second language (Workspace snapshot). The students is instructed first to assemble two main clauses (sentences (1) and (2)), then to combine them with *weil* 'because' and to pronominalize one of the subject NPs. The student failed to place *wollen* 'want' in clause-final position.

can often 're-construct' the well- or ill-formed sentences produced by students.

The design of the generator enables sentence combining in direct-manipulation style. At student request, it can join together two or more independent clauses or sentences into a larger complex or compound sentence—e.g., with one clause becoming a subordinate adverbial or relative clause within the other (as in (5); Figure 1), or by linking them together as coordinate structures (as in (6)). By dragging a function word from the word list—e.g. a relative pronoun, a subordinating or coordinating conjunction—and attaching the current clauses or sentences to it, the student can specify which sentence combination s/he wants. The generator's linguistic rule base ensures that the linguistic constraints entailed by the combination are obeyed (e.g. linear order changes, pronominalization, ellipsis).

This sentence-combining procedure is hard to realize in generators embodying the three-stage pipeline architecture described in Reiter & Dale (2000), which is not intended to deal with structural changes. Actually, COMPASS allows students not only to link trees together, but also to break them apart after making a mistake. By dragging a lexical treelet or a larger subtree away from the current overall tree, they disconnect the former from the latter. The feature composition of the nodes of the separated partial trees is immediately adapted to the constraints prevailing in the new configurations.

The above drag & drop facilities are realized by a user interface with powerful capabilities for drawing and manipulating trees interactively (Kempen 2004). Trees can be displayed with varying levels of morphosyntactic detail (e.g. showing vs. hiding structure within major phrases of a clause) and in different styles (e.g. phrase-structure vs. dependency trees). These facilities support visual grammar instruction tailored student's level of grammatical knowledge (Kempen 1999). The grammatical nomenclature in the tree diagrams and the error messages are close to that used by language teachers in traditional curricula (e.g., emphasizing grammatical FUNCTIONS; cf. Reuer 2003). Also, the 'flat' trees generated by the underlying Performance Grammar are relatively easy to understand by beginning learners of grammatical notions.

3 Current implementation

A prototype version of COMPASS has been implemented in Java, based on the Performance Grammar Workbench (PGW)—the generator described in Harbusch et al. (2006). It is intended as a software tool in support of integrated writing and grammar curricula for 10 to 14 year old elementary and secondary schoolers. The grammar and the lexicon of the system are in German; this also holds for the grammatical nomenclature in tree diagrams displayed on the screen, although users who are studying German as a second language can opt for trees with English or Dutch terminology.

When COMPASS starts, it shows three windows, called TASK, VOCABULARY, and WORKSPACE, respectively. The Task window specifies the problem to be solved in the Workspace, e.g., to construct the tree for a given sentence, to change the number and/or tense of a given sentence, or to build a few sentences describing a comic strip. The Vocabulary window lists a small set of inflected words from which the student has to choose. S/he drags the words s/he thinks are appropriate from the Vocabulary window and drops them in the Workspace. There, COMPASS displays the treelets associated with the selected word forms, enabling the student to combine them and assemble the target sentence. The sentence generator evaluates each attachment attempt and provides feedback in case of student errors (as explained in Section 2).

The exercise in Figure 1 deals with word order differences in main and subordinate clauses (the modal verb *wollen* 'want' in "second" and "final" position, respectively). By applying a mal-rule that allows verb-second in subordinate clauses, COMPASS can match the ill-formed subordinate *weil* clause assembled by the student and issue an accurate error message. The student can correct the errors by dragging the *wollen* branch to the final position.

4 Conclusion and discussion

At the time of writing, COMPASS exists only in the form of a prototype with a limited vocabulary and grammar. Together with the Psychology Department of the university of Koblenz-Landau and with teachers of German, we are designing and implementing grammar and writing exercises that are useful, attractive and motivating for the target group, and that can be tried out in the classroom. This enables us to test whether the on-line diagnostic performance of our generator-based system is at least as good as that attained by modern parser-based systems.

References

Bateman, J. A. (1997). Enabling technology for multilingual natural language generation: The KPML development environment. *Journal of Natural Language Engineering, 3,* 15—55.

Daiker, D. A., Kerek, A. & Morenberg, M. (Eds.). (1985). *Sentence Combining: A rhetorical perspective*. Carbondale: Southern Illinois University Press.

Reiter. E. & Dale, R. (2000). *Building natural language generation systems*. Cambridge UK: Cambridge University Press.

Delmonte, R., Delcloque, Ph. & Tonelli, S. (Eds.) (2004). *Procs. InSTIL/ICALL2004 Symposium on NLP and speech technologies in advanced language learning systems* (Venice, Italy). Padova: Unipress.

Harbusch, K., Kempen, G., van Breugel, C. & Koch, U. (2006). A generation-oriented workbench for Performance Grammar: Capturing linear order variability in German and Dutch. In: *Procs. 4th Internat. NLG Conference* (Sydney, Australia).

Heift, T. & Schulze, M. (Eds.) (2003). Error diagnosis and error correction in CALL. *CALICO Journal, 20(3)*. (Special issue).

Kempen, G. (1999). Visual Grammar: Multimedia for grammar and spelling instruction in primary education. In: Cameron, K.C. (Ed.). *CALL: Media, design, and applications*. Lisse: Swets & Zeitlinger.

Kempen, G. (2004). Interactive visualization of syntactic structure assembly for grammar-intensive first- and second-language instruction. In: Delmonte et al. (Eds.).

Kempen, G. & Harbusch, K. (2002). Performance Grammar: A declarative definition. In: Nijholt, A., Theune, M. & Hondorp, H. (Eds.), *Computational linguistics in the Netherlands 2001*. Amsterdam: Rodopi.

Mellon, J. C. 1969. *Transformational sentence-combining: A method for enhancing the development of syntactic fluency in english composition*. Urbana, IL: National Council of Teachers of English.

Reuer, V. (2003). Error recognition and feedback with Lexical Functional Grammar. *CALICO Journal, 20*, 497-512.

Schuurs, U. R. I. (1990). *Leren schrijven voor lezers: Het effect van drie vormen van probleemgericht schrijfonderwijs op de zinsbouwvaardigheid*. PhD Dissertation, Twente University, Enschede.

Shermis, M. D., & Burstein, J. (2003). *Automated essay scoring: A cross-disciplinary perspective*. Hillsdale, NJ: Erlbaum.

Zamorano Mansilla, J. R. (2004). Text generators, error analysis and feedback. In: Delmonte et al. (Eds.).