In: L. Appelo & F.M.G. de Jong (Eds.). *Computer-Assisted Language Learning. Proceedings of the Seventh Twente Workshop on Language Technology*. Enschede: University of Twente. [pp. 41-46]

TOWARD AN INTEGRATED SYSTEM FOR GRAMMAR, SPELLING AND WRITING INSTRUCTION

Gerard Kempen & Alice Dijkstra

Experimental and Theoretical Psychology Unit, University of Leiden
P.O. Box 9555, 2300 RB Leiden, The Netherlands
tel. +31 (71) 273834, {kempen,dijkstra}@rulfsw.LeidenUniv.nl

ABSTRACT

Orthographic accuracy is an important goal of writing instruction in primary and secondary education. If the language has a rich morphology, this goal presupposes a great deal of grammatical insight on the part of the student. However, the transfer of training from grammar instruction to writing and spelling is often very low, causing poor mastery of syntax related aspects of orthography. In this paper we argue for a written language curriculum that is strongly intertwined with initial grammar instruction. We outline the design of an integrated system of software modules that supports a unified grammar, writing and spelling curriculum. The design is largely based on existing modules, developed within our group, for processing (parsing and generating) Dutch words and sentences, and for teaching and exercising grammatical concepts.

1. SYNTAX SENSITIVE ASPECTS OF DUTCH ORTHOGRAPHY

One of the central goals of writing instruction in Dutch and Flemish primary and secondary education is orthographic accuracy. This is due to a notoriously problematic aspect of Dutch spelling. Like in French, and in contrast with English and German, the spelling rules of Dutch systematically produce homophonic but heterographic inflections. For example, the verb *verkleden* (Eng. *to dress up*) has three conjugation forms that sound the same but are spelled differently: in addition to *verkleden* which serves as infinitive and present tense plural form, there are the past tense singular *verkleedde*, the past tense plural *verkleedden*, and the inflected past participle *verklede* (I disregard some further functions of these forms). Another example is the spelling of third person present tense singular verbs ending in the phoneme /t/. *Went*, *wend* and *wendt* are pronounced identically but *went* is derived from the infinitive *wennen* (*to get used to*) whereas *wend* and *wendt* are forms of *wenden* (*to turn*). *Wend* serves as imperative, as first person singular and, in case of inversion (i.e., subject following the finite verb), as second person singular. *Wendt* is third person singular, and second person singular in clauses without inversion (subject preceding the finite verb). Problematic cases like these are not restricted to rare verbs; e.g., the /t/-spelling rules force writers to choose between *word* and *wordt* — highly frequent forms of the passive auxiliary and copula verb *worden* (*to be*, *to become*)[1].

In practice one often tries to solve such spelling problems by invoking ad hoc heuristics such as the following: "If you want to know how to spell *word(t)* in *Ik word* (*I am*), *Je wordt* (*You are*), *Word ik* (*Am I*) and *Word je*(*Are you*), then try a verb like *werken* (*to work*) where you do hear the inflectional ending /t/. Since you do not hear a /t/ in *Ik werk* (*I work*), you write *Ik word* without a *-t* ending. Similarly, one writes *Je word<u>t</u>* and *Word je* in analogy with *Je werk<u>t</u>* and *Werk je*." Such heuristics are indeed helpful but do not always yield the correct solution. For instance, *je* is ambiguous between personal and possessive pronoun (*you* versus *your*). So, the writer has to realise that in *Word(t) je moeder geholpen*? (*Is your mother being helped?*) *je* belongs to *je moeder*, and that *Werk<u>t</u> je moeder* rather than *Werk je* is the correct analogy. Of course, teachers have come up with a trick to determine whether *je* is a personal or possessive pronoun: "If one can change *je* to *jij* without making the sentence ungrammatical, it is a personal pronoun; it is a possessive pronoun if you can change it to *jouw*." However, this heuristic is not foolproof either. Although in *Word je eigen baas!* (*Be your own boss!*) one can substitute *jouw* for *je*, this does not imply *wordt* as the correct spelling: *word* is used here as imperative, and *je eigen baas* is predicate rather than subject NP.

2. INITIAL GRAMMAR INSTRUCTION

Other examples of similar import are easy to find (e.g., see Kempen, 1993). They inevitably lead to the conclusion that in order to spell homophonic but heterographic inflection forms correctly one needs to

[1] Actually, by applying grapheme-to-phoneme conversion to over 250,000 inflected forms originating from Van Dale's Dictionary of Modern Dutch (1991), we have found about 2150 sets of two or more homophonic but heterographic words. The largest set contains 6 members: *weid*, *weidt*, *weit*, *wijd*, *wijdt* and *wijt*. The number of differently spelled words in these sets amounts to about 4400.

perform at least a partial syntactic analysis of the sentence. This in turn calls for a prominent position of grammar instruction in primary and secondary school curricula. However, in spite of all the attention that is given to syntactic analysis of sentences in Dutch schools, the results of this training are very poor (e.g. see Prevaes 1994). To make things even worse, whatever parsing skills are acquired during these lessons hardly transfer to writing skills (Schuurs 1990). The causes for this regrettable situation are legion and eliminating them is anything but easy. Factors interfering with successful grammar instruction certainly include the following:

(1) the low level of motivation in students (and not seldom in teachers) due to the perception of grammar as a subject of little practical utility

(2) the poor quality of traditional paper-and-pencil parsing exercises (insufficient time for practising, late and fragmented feedback, inflexible and uninteresting sentence materials), and

(3) the lack of a transparent representation of syntactic structure.

See Pijls, Daelemans & Kempen 1987, Kempen & Jongen-Janner 1990, and Kempen 1993 for appraisals of current grammar instruction methods.

3. A UNIFIED APPROACH TO TEACHING WRITING AND PARSING SKILLS

To sum up, writing instruction is hampered by the lack of grammatical insights in students and by insufficient transfer of such insights to writing tasks, and grammar instruction seems to lack compelling arguments in support of its immediate practical usefulness.

This formulation of the two problems reveals that they could be important assets to each other's solution. Teaching writing and parsing skills as one integrated subject matter will enhance the transfer of grammatical insights to writing on the one hand, and provides a tangible argument for the practical value of grammar on the other. In what follows we will outline a unified approach to writing and grammar instruction based on various software tools created by members of our research group. As is generally recognised, present-day microcomputers offer excellent opportunities for skill acquisition. We will put emphasis on training in applying syntax sensitive rules of Dutch orthography and on parsing exercises.

4. CONTROLLED WRITING AND PARSING CONTROLLED TEXTS

It would be utterly unrealistic to propose a CALL system consisting of the following components:

(1) a full-fledged 'grammar checker' that accurately diagnoses the grammatical quality of texts about arbitrary topics typed into a word-processor file by, for example, fifth- or sixth-graders

(2) an 'intelligent tutoring system' that, based on the checker's diagnoses, infers lacunae in the students' grammatical knowledge ('student modelling') as well as customised remedial grammar curricula, and

(3) a varied collection of instruction modules and exercises for inclusion in the curricula.

In view of prevailing hardware and software limitations we have set ourselves a modest target which we believe could still make for considerable improvement.

First and foremost, the writing tasks to be performed by the students will not be unconstrained creative writing assignments. In order to keep error diagnosis at an acceptable level of accuracy, we will use 'controlled writing' tasks, e.g. of the 'sentence combining' variety. Paper-and-pencil versions of this task that have been applied successfully in the United States (Dauker, Kerek & Morenberg 1985, 1986). The basic idea is as follows. Students are presented with a sequence of short clauses that each express an atomic proposition. Together the propositions make up a little story or article. By transforming the short clauses and combining them into longer sentences, the students should produce a coherent and fluent piece of text. A simpler alternative — but even farther removed from unconstrained creative writing — would be to display on the screen a varied selection of words (citation forms) whose conjugation or declension in sentential contexts involves difficulties of the kind discussed in Section 1. The students are then instructed to make up arbitrary but grammatical sentences using the given word materials.

Secondly, we do not propose to have such 'controlled texts' analyzed by a grammar checker. Instead we will use a stripped and adjusted version of the syntax sensitive spelling checker for Dutch that was designed and implemented by Vosse (1992, 1993, forthcoming). The central component of this system is a Generalized LR(1) shift-reduce parser operating on the basis of an augmented context-free grammar. It was specifically designed for the detection and correction of syntax dependent spelling errors as described in Section 1. These errors usually surface as mismatches between morphosyntactic features within a constituent. For instance, the misspelling *Ik wordt* can be analyzed as missing agreement between subject (first person) and finite verb (second or third person). Vosse created a robust feature unification mechanism that, rather than giving up in case of unification failure, attempts to pin-point the missing agreement(s) and to find the 'cheapest' way of restoring correctness. The implied changes to one or more words of the input sentence are then displayed as a suggested correction. For details see Vosse's forthcoming dissertation.

A PC version of the parser with the original vocabulary of over 250,000 entries and over 500 grammar rules has been tested extensively on various types of texts. Its performance in terms of accuracy of agreement error detection and of speed is certainly unsatisfactory for unconstrained texts. But we estimate that a stripped version with a smaller vocabulary and a simplified grammar tuned to the sentence materials in the controlled writing exercises will be able to reach an acceptable level of performance.

The third restriction we wish to impose on the system has been inspired by our experiences with an earlier 'educational word-processor' (Vosse 1989). Parser performance degrades drastically when punctuation marks are missing or incorrect so that accurate segmentation of the text into sentences is impossible. Since punctuation by fifth- and sixth-graders typically leaves much to be desired, it will be necessary to run a text segmentation routine, e.g. the one in Vosse's system, before activating the parser. Also, we need a user interface module that helps students to polish up punctuation before sending the text to the parser.

5. TRAINING SYNTAX SENSITIVE SPELLING RULES

In the framework of a CALL program it would obviously make no sense to have the system repair all misspellings automatically. Instead, misspellings and their diagnosis by the parser should activate a teaching module capable of

(1) explaining the nature of the error in terms of pertinent morphosyntactic rules

(2) offering opportunities for training the rules that apparently are beyond the student's mastery, and

(3) eliciting from the student a correction of the misspelling, and checking its quality.

During each of these steps further gaps in the student's knowledge or skill may come to light, e.g. lack of understanding the grammatical concepts used in error explanations, or insufficient rule application skills. Since each such gap may cause activation of another module, potentially elaborate hierarchies of calls to explanation and teaching modules may come about. In order to avoid the risk of students getting lost in a forest of low-level details and forgetting about the top-level writing task, we will need to devise a more flexible control structure. We leave this didactic problem as an important system design issue and, in the rest of this paper, concentrate on the nature of the teaching and exercise modules instead.

6. EXPLAINING ERRORS AND GRAMMAR RULES

The system must be able to handle two different cases. A spelling error may lead to a non-existent word, or to an existing word that causes the sentence to be syntactically ungrammatical, e.g. because of an agreement error. In the first case the student may need help to determine the correct spelling of the intended word (cf. Section 8). In the second case the student may first need guidance in locating the misspelled word that causes the agreement error. For example, the parser may inform the student that an agreement error has been found between the subject and the finite verb. To understand the meaning, and — more importantly — to know the course of action to be taken to remedy this error, the student should be able to obtain more specific information about the type of error, and information pertaining to the grammar rule(s) involved.

A hypertext-like database system, as was used in a system supporting Dutch writers of English (Dijkstra & Sijtsma 1992; De Ruiter & Dijkstra 1993), can help the student find the appropriate information. In order for the student not to get lost in hyperspace, the grammatical information in this database should be very well organized. The Dutch curriculum for initial grammar instruction developed by Jongen-Janner, Reimann-Pijls & Kempen (1991) seems ideally suited for this purpose. It introduces and explains virtually the complete grammatical terminology needed to describe the grammatical structure of non-compounded sentences (almost 100 terms). The operational definitions of these terms adhere to the format of a genetic graph; that is, every new concept is defined in terms of already known concepts. The 'dependency links' thus created are easily converted to hypertext links.

An error diagnosis produced by the parser can direct the student to the appropriate information in the hypertext database. If the parser is incapable of determining the exact nature of the error, the student himself may be asked to build a representation of the syntactic structure of the intended sentence. By mouse manipulation he could indicate the words and constituents in the sentence and associate with them the names of parts of speech, word groups and grammatical functions. (Large parts of the user interface of TAALTRIS, the computer assisted grammar exercises described below in Section 7, are suited to this purpose.) On the basis of this structure the parser will then be able to determine the appropriate error diagnosis.

Once the student has located and understood the error he has made, he should be urged to correct it. If he does not know how, a spelling tutor can be invoked to help him out (cf. Section 8).

In order to provide the student with a better insight into the grammatical organization of sentences, their structure can be displayed two-dimensionally, e.g. in the form of trees, of directed acyclic graphs (DAGs), or of 'recursive temples' designed by Diesveld & Kempen (1993). To this purpose we have developed various graphical tools (Bos & Hensgens 1989).

7. ATTRACTIVE COMPUTER ASSISTED GRAMMAR EXERCISES

Grammar instruction cannot restrict itself to conveying grammatical *insights* to the students. An equally important goal is *skill* acquisition: students should reach a sufficiently high level of proficiency in applying the operational definitions of the grammatical concepts, e.g. in identifying parts-of-speech (POS, i.e. word categories) and syntactic constituents of various types, in recognizing their morpho-syntactic properties, and in constructing utterances that instantiate such concepts. Any grammar curriculum should therefore provide extensive and attractive materials to exercise these skills.

In an attempt to maximize the students' extrinsic motivation to engage in grammar exercises, we have designed an adaptation of the well-know computer game of TETRIS (Diesveld & Kempen 1993; we baptized it TAALTRIS: Du. *taal* = Eng. *language*).

The computer screen displays a box with a row of bricks on the bottom. A word is printed on every brick; the words together make up a sentence. The top of the screen shows another set of bricks with part-of-speech names of printed on them. The game's objective is to pair up the words with their word classes by directing the POS bricks to the word bricks as fast as possible. When the student does not succeed in completing a sentence in the time span allotted, a new sentence appears on top of the previous one. The student then proceeds by assigning parts-of-speech to the words of this new sentence. If he succeeds in time, the second sentence disappears and the student may now complete POS assignment of the earlier sentence. In case of bad performance, the box will gradually get filled up with word bricks, causing the student to lose. The game, implemented by Peter Diesveld, contains various other gadgets and trimmings similar to those used in popular computer games. Currently we work with two versions: in addition to the one for POS assignment, we have another one where the word bricks have been replaced by *word group* bricks, and the POS bricks by bricks with names of *grammatical functions* printed on them (subject, direct object, finite verb, adverbial modifier, etc.). The goal of this version is to acquire skill in identifying the grammatical functions of major phrases at clause level.

In an ongoing evaluation experiment with 60 sixth-graders in two primary schools we have observed that students find the game highly attractive. More importantly, their proficiency in POS assignment and in identifying grammatical functions seems to improve markedly.

The sentence materials selected for the TAALTRIS grammar game were picked from children's books or made up by teachers. To increase the involvement and motivation of the students, they can be encouraged to turn the sentences they generated during controlled writing exercises (Section 4), into exercise material for fellow students. For this purpose, some of the TAALTRIS components will be re-usable. However, the parts of speech and grammatical function assignments by the students must now be checked by the parser.

8. A SPELLING TUTOR FOR DUTCH

To determine the inflectionally and orthographically correct form of intended words, a spelling tutor will be included. For the category of Dutch *verbs*, such a tutor has already been developed: HET SPELRAAM (Kempen & Janner 1990). It operates on the basis of morpho-syntactic knowledge built into the program and is equipped with error diagnostic tools based on that knowledge (see Kempen 1992 and Bos 1994 for details).

HET SPELRAAM includes a spelling algorithm consisting of two stages. The first, so-called morpho-syntactic stage generates a formula specifying the prefix and/or suffix to be attached to the verb's stem for the particular conjugation. An example of such a formula is *(ge+)stem+d/t* for past participles, equivalent to the English *stem+ed* formula. The substring *(ge+)* indicates an optional prefix *ge*, while *d/t* indicates a choice between the homophonic endings *-d* and *-t*. These choices have to be made in the second, so-called orthographic stage (see below).

In order to find the appropriate formula, the student traverses a decision tree by answering a series of questions concerning morphological features of the desired verb form and the syntactic structure of the sentence. Some examples are the following: Is the required verb form finite or non-finite? Is the tense present or past? Does the sentence contain subject-verb inversion? Is the number of the subject singular or plural? When the student answers these questions correctly, he arrives at the appropriate formula. If not, he should be given appropriate feedback and, if necessary, be guided towards the relevant information in the grammar database described in Section 6.

In the second stage of the spelling algorithm, the formula obtained in the first stage must be applied to the verb's citation form (the infinitive). This involves a maximum of three steps, each requiring some string manipulations according to morphological and/or phonological rules involved. First, the verb's stem must be determined. Second, the prefix and/or suffix(es) specified in the formula must be added to the verb's stem, and some additional string manipulations must be performed in order to conform to certain phonological principles, e.g. changing the stem's final consonant *v* into *f*. A third and final step is only needed in case of inflected participles, which call for an additional *-e* ending and possibly some further alterations to the end result of the second step. In this stage too, HET SPELRAAM checks the actions performed by the student and provides appropriate feedback and help if necessary.

For the system proposed in this paper, HET SPELRAAM should be extended to include algorithms for inflecting and spelling adjectives and nouns. Moreover, the new version should communicate with the parser so as to be able to check the student's responses during the morpho-syntactic stage.

9. SYSTEM ARCHITECTURE, STUDENT INTERFACE, AND STUDENT MODELING

The software components we have proposed in the previous Sections should all be embedded and integrated in a larger system. The central module will be a simple text processing system from which the various components are called. Furthermore, the system will have to provide for transfer of information between the modules.

A relatively simple but very useful student modeling tool may be based on automatic registration of errors made by individual students or by homogeneous groups of students. Such information will help the parser and spelling tutor to find correct error diagnoses, and will guide the system in selecting proper grammatical feedback and remedial grammar exercises.

* * *

In this paper we have emphasized the role that various individual modules can play in the proposed computer assisted system for written language and grammar instruction. Integrating these modules and adapting the existing versions to their new environment will not be an easy task. Nevertheless we are convinced that building the system will be rewarding in view of the expected increase in learning speed and teaching efficiency.

ACKNOWLEDGEMENT

We are indebted to Theo Vosse, Elena Jongen-Janner and Peter Diesveld for their invaluable contributions to the design and implementation of current versions of the modules described in this paper.

REFERENCES

Bos, E. & Hensgens, J. (1989). Directe manipulatie van ontleedbomen. In: F. Pijls & J. Sandberg (Eds.) *De computer als expert en didacticus*. Muiderberg: Coutinho.

Bos, E. (in press). Error diagnosis in a tutoring system for the conjugation and spelling of Dutch verbs. *Computers in Human Behavior*.

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

Dauker, D.A., Kerek, A. & Morenberg, M. (1986). *The writer's options: combining to composing*. New York: Harper & Row.

Diesveld, P & Kempen, G. (1993). Zinnen als bouwwerken: computerprogramma's voor grammatica-oefeningen. *MOER, Tijdschrift voor onderwijs in het Nederlands*, *1993-4*, 130-138.

Dijkstra, A. & Sijtsma, W. (1992). *TRICS: an editorial support tool for Dutch authors writing in English*. Leiden, RUL: SPIN-MMC report.

Jongen-Janner, E., Reimann-Pijls, F. & Kempen, G. (1991). *Het Taalbos*. (Tekstboek en Werkboek). Interne Rapporten, NICI, KU Nijmegen.

Kempen, G. (1992). Language technology and language instruction: computational diagnosis of word level errors. In: M.L. Swartz & M. Yazdani (Eds.) *Intelligent tutoring systems for foreign language learning. The bridge to international communication*. Berlin: Springer.

Kempen, G. (1993). *Spraakkunst als bouwkunst*. Inaugural Address, University of Leiden, March 19, 1993.

Kempen, G. & Jongen-Janner, E. (1990). Naar een flexibele methode voor algoritmisch grammatica- en spellingonderwijs. *Pedagogisch Tijdschrift, 15*, 280-289.

Pijls, F., Daelemans, W. & Kempen, G. (1987). Artificial Intelligence tools for grammar and spelling instruction. *Instructional Science, 16*, 319-336.

Prevaes, M. (1994). *Het oplossen van grammaticale problemen door negenjarigen*. Doctoral dissertation, University of Amsterdam.

Ruiter, J.P. de & Dijkstra, A. (1993). *TRICS: Functional and technical design*. Leiden, RUL: SPIN-MMC report.

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

Vosse, Th.G. (1989). Een slimme tekstverwerker voor grammaticale ondersteuning van het schrijfonderwijs. In: F. Pijls & J. Sandberg (Eds.) *De computer als expert en didacticus*. Muiderberg: Coutinho.

Vosse, Th.G. (1992). Detecting and correcting morpho-syntactic errors in real texts. In: *Proceedings of the Third Conference on Applied Natural Language Processing*. ACL.

Vosse, Th.G. (1993). Robust GLR Parsing for Grammar-based Spelling Correction. In: K. Sikkel & A. Nijholt (Eds.) *Natural Language Parsing: Methods and Formalisms* (Proceedings of the Sixth Twente Workshop on Language Technology). Enschede: University of Twente.

Vosse, Th.G. (1994, in preparation). Doctoral dissertation, University of Leiden.