Language choice models for microplanning and readability

Sandra Williams
Department of Computing Science
University of Aberdeen
Aberdeen AB24 3UE, UK
swilliam@csd.abdn.ac.uk

Abstract

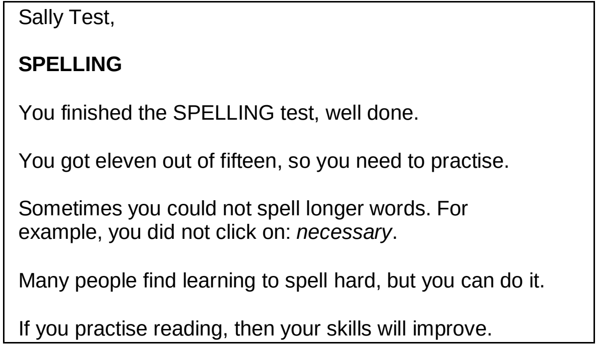
This paper describes the construction of language choice models for the microplanning of discourse relations in a Natural Language Generation system that attempts to generate appropriate texts for users with varying levels of literacy. The models consist of constraint satisfaction problem graphs that have been derived from the results of a corpus analysis. The corpus that the models are based on was written for good readers. We adapted the models for poor readers by allowing certain constraints to be tightened, based on psycholinguistic evidence. We describe how the design of microplanner is evolving. We discuss the compromises involved in generating more readable textual output and implications of our design for NLG architectures. Finally we describe plans for future work.

1 Introduction

Generator for Individual Reading Levels (GIRL) is a Natural Language Generation (NLG) system that generates feedback reports for adults during a web-based literacy assessment. The inputs to GIRL are answers to questions in a literacy assessment. GIRL currently generates a feedback report after each of eight skill-based tests in the assessment. An example output report, generated after the spelling test, is shown in Figure 1.

GIRL is being developed with the aim of tailoring its output texts to the individual reading skills of readers. Our particular focus is on adults who have poor reading skills due to a number of reasons including missed school, dyslexia, poor eyesight, memory problems, etc. Poor literacy is a major problem in the UK where up to one fifth of the adult population is functionally illiterate (Moser 1999).

Using Kintsch and Vipond's (1979) definition, we relate readability to performance on the reading task (i.e. reading speed, ability to answer comprehension questions and ability to recall content). We measured the first two of these in preliminary experiments that tested outputs from GIRL on both good and bad readers (Williams et al. 2003).



Sally Test,

SPELLING

You finished the SPELLING test, well done.

You got eleven out of fifteen, so you need to practise.

Sometimes you could not spell longer words. For example, you did not click on: *necessary*.

Many people find learning to spell hard, but you can do it.

If you practise reading, then your skills will improve.

Figure 1. A feedback report generated by GIRL

Our research is focused on decisions GIRL makes at the discourse level. A previous project, PSET (Devlin and Tait 1998, Devlin et al. 2000), has already made some progress towards lexical-level and syntax-level simplifications for poor readers. In GIRL, it is at the discourse level that choices are made that affect sentence length and selection of discourse cue phrases (phrases that render discourse relations explicit to the reader, e.g. 'for example', 'so' and 'if', in Figure 1). These choices are made in a module called the microplanner (see Reiter and Dale 2000).

The inputs to the microplanner are a model of a user's reading ability and a tree-structured document plan (Reiter and Dale 2000) that includes discourse relations. In GIRL, discourse relations are schemas arranged in a discourse tree structure. Each schema has slots for semantic roles filled by daughter text spans, or daughter relations. For instance, the *condition* relation has two semantic roles: a condition and a consequent. Figure 2 shows a discourse relation tree structure with its corresponding schema. The root relation, R1, is a *concession* (type: concession), with one daughter relation, R2, filling the 'concession' slot and a text span daughter, S1, filling the 'statement' slot. R2 is a condition relation with two text span daughters: S3 filling the 'condition' slot and S2 the 'consequent' slot.

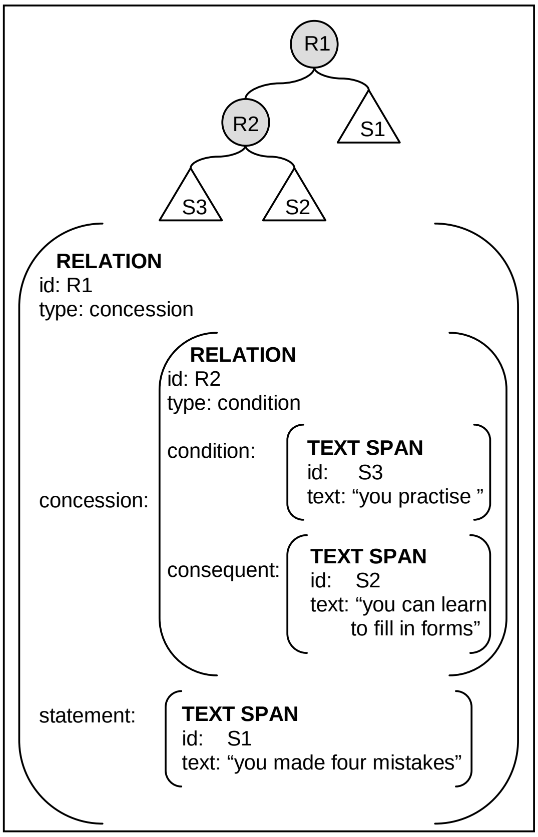


Figure 2. Discourse relation tree structure and schema.

The task of GIRL's microplanner is to decide on the ordering of the daughters, how they should be packed into sentences (aggregation), whether there should be punctuation between the daughters, whether discourse cue phrases should be present and, if so, which ones and where they should be placed. The microplanner will ultimately adapt the choices it makes to the reading level of individual users (readers) from user models built from users' answers to up to ninety questions from a literacy test. Our current implementation only considers two generic types of user - "good readers" and "bad readers".

Suppose the input to the microplanner is a discourse plan containing the discourse relation tree in Figure 2. It should be able to calculate that this could be generated in a number of different ways. Just a few of them are:

- You made four mistakes. But you can learn to fill in forms if you practise.
- Although you made four mistakes, you can learn to fill in forms ... just as soon as you practise.
- You made four mistakes. But if you practise, you can learn to fill in forms.
- If you practise, you can learn to fill in forms. You made four mistakes, though.

and it should be able to choose which of these is the most appropriate for poor readers.

The remainder of this paper describes what we believe is a novel approach to building language choice models for microplanning. We explain how these models evolved (section 2) and the implications of this design (section 3). Section 4 draws conclusions from the current work and outlines our plans for future work.

2 Constructing the microplanner

This section describes the stages in the construction of the microplanner. Each stage is based on empirical evidence. Firstly, we acquired knowledge about how human writers linguistically realise specific discourse relations by carrying out a corpus analysis (see Williams and Reiter 2003). Secondly, we selected the best method for representing this knowledge and built choice models from the corpus analysis data. Then, because the corpus was written for good readers, we had to adapt the models for poor readers. For this, we used results from psycholinguistic studies, including results from our own preliminary experiments (see Williams et al. 2003). Finally, these individual parts were combined to produce the finished microplanner.

2.1 Reconfiguring our corpus analysis results

We analysed seven discourse relations (Williams and Reiter 2003), including *concession*, *condition*, *elaboration-additional*, *evaluation*, *example*, *reason* and *restatement*, using the RST Discourse Treebank corpus (Carlson et al. 2002). We analysed one hundred instances of each relation noting the following six features:

- **L1**: length of the first text span (in words).
- **L2**: length of the second text span (in words).
- **O**: ordering of the text spans.
- **Ps**: position(s) of discourse cue phrase(s).
- **P**: between-text-span punctuation.
- **C**: discourse cue phrase(s).

An example to demonstrate these features is the *concession* relation in the last example given above: *"If you practise, you can learn to fill in forms. You made four mistakes, though."* Here, **L1** is ten words (this includes the whole of the *condition* daughter), **L2** is five words, **O** is concession-statement, **Ps** is after the statement, **P** is a full stop and **C** is "though".

These features were chosen on the basis of previous work (Moser and Moore 1996) and because they influence sentence length and lexical choice which are known to be important factors in readability. The analysis revealed some of the ways in which human authors select these features when writing for good readers. These provided a partial specification for modelling discourse-level choices that should be available in an

NLG system. Furthermore the analysis demonstrated that the features are interdependent.

The results from our corpus analysis (Williams and Reiter 2003) were simplified. The numbers of values for some features were cut down by re-classifying them as members of a smaller number of categories. Length became either “long” or “short”. The data for each relation was split into two, so that roughly half the L1 instances fell into the “short” category (e.g. for *concession*, short = 1-15 words, long = >15 words). Between-text-span punctuation was divided into just three categories: none, non-sentence-breaking, and sentence-breaking. The *restatement* relation was an exception because it had such a large proportion of open-parentheses (62%) that an extra category was created. In restatement, it seems that punctuation is often used instead of a cue phrase to signal the relation. The cue phrase feature was left with larger numbers of values to provide GIRL with the maximum number of choices for lexical selection.

The data was reconfigured as sets of 6-tuples. Each represents a set of values for one instance of a relation: i.e. <L1,L2,O,Ps,P,C>. For instance, the *concession* relation described above would be represented as <short, short, concession-statement, after_statement, full stop, “though”>. We thus created seven hundred 6-tuples in total, one hundred per relation. For each relation, these were sorted, duplicates were counted and superfluous duplicates removed. Of the resulting unique 6-tuples, some were rejected and are not used in the current language choice models. For example, in the *concession* choice model forty-six unique 6-tuples cover 100% of the corpus data and sixteen were rejected, resulting in a coverage of 75%. For *condition*, forty-seven unique 6-tuples cover 100% but only twenty-six were included and these cover 72%.

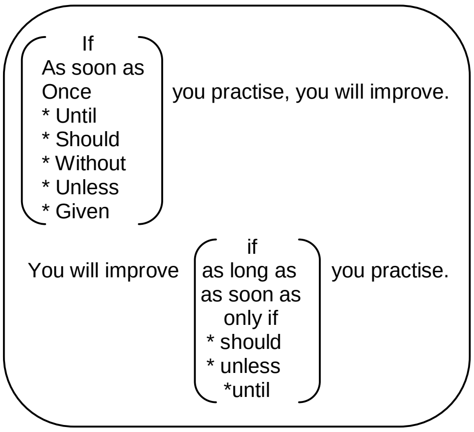


Figure 3. Some cue phrases, with those not in the current language models marked with asterisks

The reason why some tuples were rejected is because GIRL’s present shallow approaches to syntactic and semantic processing cannot generate them. It cannot currently generate embedded discourse text spans, nor can it generate discourse cue phrases in mid-text-span positions. Both of these would require the implementation of deeper syntactic processing. Certain 6-tuples contain discourse cue phrases that would not make sense when generated unless we implement deeper semantic processing. Figure 3 shows some examples of these. Cue phrases marked with asterisks have been rejected from the current language models because they require deeper processing.

Our current method for reconfiguring the data is manual, using existing spreadsheet, database and statistics packages. We are investigating how it could be automated, given that some decisions, such as which 6-tuples to reject, require human judgement.

2.2 Building CSP graphs for good readers

Having reconfigured the results of our corpus analysis, we searched for the best way to model the choices they represent. We tried exploring both discriminant analysis statistics and machine learning of decision trees in attempts to identify which feature(s) would most clearly divide the data into groups. For most discourse relations, the positions of discourse cue phrases were the most discriminating features.

The most crucial characteristic of the choice models we were attempting to build was that they should reflect the interdependencies of the features found in the corpus analysis. For instance, in most relations the selection of between-span punctuation is dependent on the length of the first text span. For some relations (not all), this means that as the first text span gets longer, the between-span punctuation tends to change from no punctuation, to comma, to full stop. Similarly, the selection of punctuation depends on the order of text spans, particularly with the *condition* relation. If the order is condition-consequent, there tends to be a comma between text spans, if the order is consequent-condition, there is often no punctuation. And so on with interdependencies between all the other features.

The best representation we have found to date that fits this requirement is constraint satisfaction problem (CSP) graphs. Power (2000) demonstrated that CSPs could be used to map rhetorical structures (i.e. discourse relation trees) to text structures (paragraphs, sentences, etc.). Our task is similar to Power’s, but we emphasise different processes, such as cue phrase choice, our choice models are based on empirical evidence, and we have the additional criteria that the representations should be adaptable for different reading abilities. It turned out that CSP graphs were ideal for this purpose, since we exploit CSP’s notion of ‘tightening’ the constraints in our solution for adapting the models for poor readers (see section 2.3).

We used the Java Constraint Library (JCL 2.1) from the Artificial Intelligence Laboratory at the Swiss Federal Institute of Technology in Lausanne (Torrens 2002) which we found to be portable, relatively bug-free and easy to plug straight into our system which is written entirely in Java.

We built computer models representing the six key features of discourse relations and their interdependent values. One CSP graph was built for each of the seven discourse relations. The structure of the graphs is exactly the same for each relation with six nodes and fifteen connections linking every node to all the others. This structure is illustrated in figure 4.

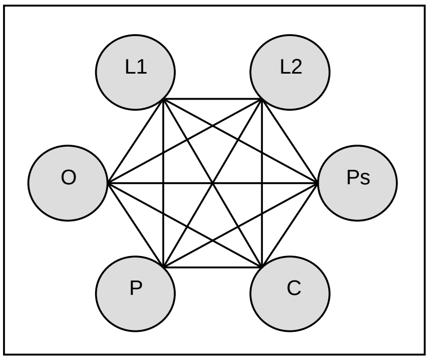


Figure 4. CSP graph representing a discourse relation.

The nodes in the graph in figure 4 are CSP domain variables. Each represents one of the six features. The numbers of values for each node varies for each relation. Constraints between the variables were represented as "good lists". Both values and constraints were coded directly from the 6-tuple data. Good lists contain pairs of values that are "legal" for two variables. For instance, a connection between L1 and P might contain the pair <*short, non-sentence-breaking*> in its good list, meaning: if the length of the first text span in the relation is short, put non-sentence-breaking punctuation, such as a comma, between the text spans. The numbers of pairs in the "good lists" attached to each of the fifteen connections varies for each relation.

We used pairs of "legal" values in the CSP good lists because the corpus analysis is too small to predict the probabilities of triples. We are currently working on expanding the size of our corpus analysis. We wanted the CSP graphs to generate solutions that gave as good a coverage of the 6-tuples included in the models as possible, but we did not want to overgenerate instances that did not occur in the analysis. This required delicate balancing of the two factors.

2.3 Adapting the models for poor readers based on psycholinguistic evidence

The language choice models were adapted for poor readers by tightening the constraints. We studied the psycholinguistic and educational literature to determine how they should be tightened. We also carried out preliminary experiments of our own (Williams et al. 2003) which indicated that certain discourse-level features affect readability for poor readers more than good readers. Selecting more common discourse cue phrases and the placing punctuation between discourse segments were both particularly helpful for poor readers.

Existing psycholinguistic research on reading has little to say about adults with poor literacy. It has tended to focus on proficient adult readers (University students), rather than on the problems of adult learner readers. Where it has investigated the development of reading skills, it has tended to focus on children, rather than adults. Educationalists maintain that the reading skill profiles of adults with poor literacy are different from those of children. 'Normal' children tend to develop reading skills evenly, whereas adults who are functionally illiterate tend to have developed unevenly (Strucker 1997). Yet another problem is that it tends to focus on single words, single sentences, or pairs of sentences, that are presented to a reader out-of-context, rather than in multiple-sentence documents.

There are some exceptions, however. Devlin and Tait (1998) found that the readability of newspaper texts was increased for seven out of ten aphasic readers when they replaced infrequent words with more frequent synonyms. Leijten and Van Waes (2001) reported that elderly readers' comprehension and recall improved when they were presented with causal discourse structures containing explicit discourse cue phrases and explicit headings. Degand et al. (1999) observed that removal of even a few cue phrases affects comprehension and recall of the entire content. The last two studies were with adult readers from the general public with (presumably) varying levels of reading ability.

To sum up, use of cue phrases, selection of common cue phrases and use of between-span punctuation all seem to help bad readers. We therefore chose to tighten the constraints to favour solutions with these features.

Frequencies for cue phrases were obained from a part-of-speech (POS) search (Aston and Burnard 1998) in the 100 million word British National Corpus. Phrases like 'for example' are annotated with a single part-of-speech in the BNC. Some results are shown in Table 1. Cue phrases do not all have the same POS, and they are not, of course, exact synonyms, so it is not always possible to substitute one for another even if both are from the same relation. 'Such as' can not always be substituted for 'for instance', but 'for example' is a close synonym and it is possible to do a substitution.

We tightened constraints, where possible, to favour words that occur in the Dolch lists used by adult literacy tutors. These list the most commonly occurring function words that beginner readers are taught to sight read.

Another danger with substituting common phrases for less common ones is that the most common phrases are also the most ambiguous. The cue phrases ‘but’ and ‘and’ both occurred in four relations (*concession*, *elaboration-additional*, *evaluation* and *reason*) out of seven in the corpus analysis and these are relations with very different meanings. These problems require further investigation.

Cue phrase	BNC freq.	Dolch list
although	42,758	-
and	2,615,144	yes
because	83,181	yes
but	443,164	yes
for example	23,643	yes
for instance	7,344	-
if	230,892	yes
still	67,106	-
though	33,337	-

Table 1. Cue, BNC frequency & Dolch list presence.

2.4 Putting it all together – the microplanner

Figure 4 shows the main components of the microplanner. The inputs are a model of the user’s reading ability (marked ‘user model’) and a document plan containing discourse relation trees, marked ‘DocPlan’. Both are built by system modules occurring earlier than the microplanner in the processing sequence. The document plan in figure 4 is the same as shown above in figure 2. Working bottom-up, a CSP graph for the current relation is retrieved from the CSP graph knowledge base and the constraints are tightened or relaxed according to the user model. The CSP Solver (Torrens 2002) then uses simple backtracking search to find all solutions for the relation. The solutions found by the CSP Solver are passed through a filter which currently picks the most frequently occurring one for good readers and the one with overall shortest sentences for poor readers. The output is a schema that the next module of GIRL uses to construct messages.

It does not always output the most coherent solution. For instance, the output shown in figure 5 would result in a final output of *“You made four mistakes. But if you practise, you can learn to fill in forms”*. Adjacent discourse cue phrases do not improve coherency. The microplanner is still under development, however, future improvements, possibly including backtracking, will improve readability, possibly including coherence considerations, such as focus and reference.

3 Discussion

Additional functionality would need to be added to the ‘filter’ module to choose solutions that optimise discourse coherence. Additional nodes might be required in the constraint graphs. The simple string content of discourse relations would have to be replaced by semantic representations. If it were, the simple pipeline architecture would no longer be appropriate, since it currently depends on knowing the final length of the strings.

On the other hand, when generating text for bad readers, we might have to sacrifice some of these, since they might impact on readability. Ellipsis, for instance, may not be good for bad readers. Ellipsis is one way that conciseness can be achieved during aggregation. Current opinion in the NLG community is that aggregation for conciseness is ‘a good thing’. Reape and Mellish (1999) even suggest that an NLG system should ‘aggregate whenever possible’. But conciseness may be less comprehensible for poor readers. The sentences in A, below, could be aggregated as in B.

A. *Spelling is hard. But spelling is important.*

B. *Spelling is hard but important.*

However, in B a single sentence is longer and the cognitive load for poor readers in working out the ellipse could be higher. A little repetition and redundancy might actually turn out to be beneficial!

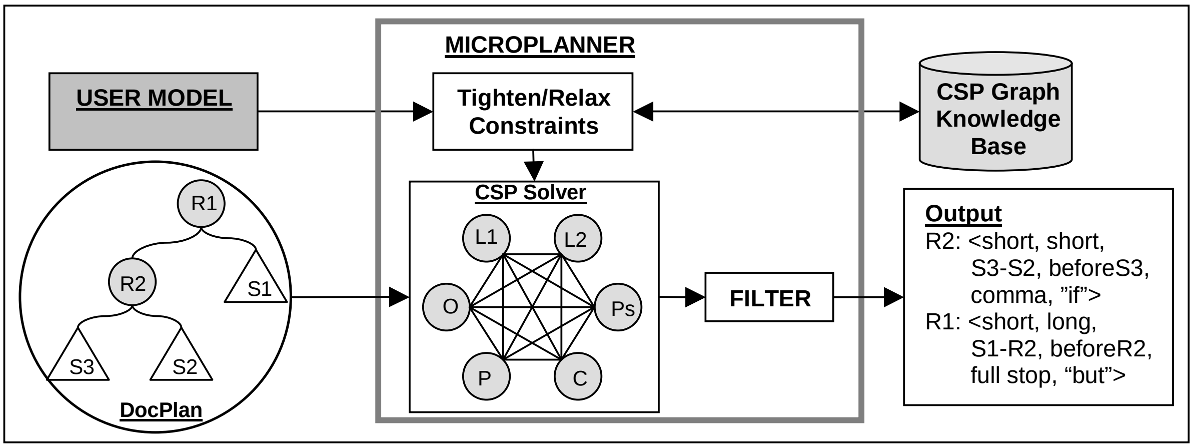

MICROPLANNER
USER MODEL
Tighten/Relax Constraints
CSP Graph Knowledge Base
CSP Solver
R1
R2
S1
S3
S2
DocPlan
L1
L2
O
Ps
P
C
FILTER
Output
R2: <short, short, S3-S2, beforeS3, comma, ”if”>
R1: <short, long, S1-R2, beforeR2, full stop, “but”>

Figure 5. The microplanner

4 Conclusions and future work

This paper described how we used the results of a corpus analysis to build language choice models for a microplanner. We discussed the creation of constraint satisfaction problem graphs for our default “good reader” models and how we adapted the models for poor readers. Our “poor reader” models are based on psycholinguistic evidence, including evidence from our own preliminary experiments. We discussed some of the compromises involved in generating more readable textual output and the impacts that further development could have on GIRL’s architecture.

Plans for future work include expanding the size of our corpus analysis and automating at least some of the analysis and data reconfiguration. We plan further development of the microplanner to prevent incoherent solutions being generated. Further on, we plan to take discourse coherence considerations into account.

We have plans to carry out additional reading experiments with good and bad readers to investigate whether the constraints we tighten to adjust the language models for poor readers actually produce more readable results. We will generate texts under the default “good reader” models and under the constrained, poor reader, models. We will measure reading speeds and comprehension as in our preliminary experiment. (Williams et al. 2003). We predict that, as we found then, good readers will perform equally well on both models and poor readers will perform better on the constrained models. We will also carry out user satisfaction evaluations and carry out evaluation surveys with professional basic skills (adult literacy) tutors.

References

Guy Aston and Lou Burnard. 1998. *The BNC Handbook: Exploring the British National Corpus with SARA.* Edinburgh University Press.

Lynn Carlson, D. Marcu, and M Okurowski. 2002. Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory. Kuppevelt and Smith (eds.) *Current Directions in Discourse and Dialogue,* Kluwer.

Liesbeth Degand, N. Lefèvre and Y. Bestgen. 1999. The impact of connectives and anaphoric expressions on expository discourse comprehension. *Document Design: Journal of Research and Problem Solving in Organizational Communication,* 1 pp. 39-51.

Siobhan Devlin and John Tait. 1998. The Use of a Psycholinguistic Database in the Simplification of Text for Aphasic Readers. *Linguistic Databases*. J. Nerbonne (ed.) CSLI Publications.

Siobhan Devlin, Y. Canning, J. Tait, J. Carroll, G. Minnen and D. Pearce. 2000. An AAC aid for aphasic people with reading difficulties. *Proceeding of the 9^th^ Biennial Conference of the International Society for Augmentative and Alternative Communication*.

Walter Kintsch and Douglas Vipond. 1979. Reading Comprehension and Readability in Educational Practice and Psychological Theory. L.Nilsson (ed.) *Perspectives on Memory Research*. Lawrence Erlbaum.

Mariëlle Leijten and Luuk Van Waes. 2001.The impact of text structure and linguistic markers on the text comprehension of elderly people. W. Spooren and L. van Waes (eds.) Proceedings of *Multidisciplinary Approaches to Discourse*.

Claus Moser. 1999. Improving literacy and numeracy: a fresh start. Report of the working group chaired by Sir Claus Moser.

Megan Moser and Johanna Moore. 1996. On the correlation of cues with discourse structure: results from a corpus study. Unpublished manuscript.

Richard Power. 2000. Mapping Rhetorical Structures to Text Structures by Constraint Satisfaction. Information Technology Research Institute, Technical Report ITRI-00-01, University of Brighton.

Mike Reape and Chris Mellish. 1999. Just what is aggregation anyway? Proceedings of the Seventh European Workshop on Natural Language Generation.

Ehud Reiter and Robert Dale. 2000. *Building Natural-Language Generation Systems*. Cambridge University Press.

John Strucker. 1997. What silent reading tests alone can’t tell you: two case studies in adult reading differences. *Focus on Basics*, Vol. 1B, National Center for the Study of Adult Learning and Literacy (NCSALL), Harvard University.

Marc Torrens. 2002. Java Constraint Library 2.1. Artificial Intelligence Laboratory, Swiss Federal Institute of Technology. GNU Lesser Public Licence.

Sandra Williams and Ehud Reiter. 2003. A corpus analysis of discourse relations for Natural Language Generation. To appear in proceedings of Corpus Linguistics 2003.

Sandra Williams, Ehud Reiter and Liesl Osman. 2003. Experiments with discourse-level choices and readability. To appear in proceedings of the 9th European Workshop on Natural Language Generation.