Adapting wording to layout

Richard Power

1 Introduction

Most work on Natural Language Generation (NLG) has focussed on the purely linguistic part of the problem — getting the words right. Issues of punctuation and formatting are addressed minimally, if at all. Of course there is nothing wrong in tackling a restricted research problem, rather than trying to solve everything at once. However, if the objective in the long term is to develop NLG systems that can be used for the commercial production of technical documents, issues of layout can no longer be ignored.

This point may not be immediately obvious. Why should a NLG system not simply produce an unformatted text file that can be loaded into any word processor and formatted by hand? One reason is that automatic formatting saves time. Another is that versions might be generated in many languages; since correct formatting requires comprehension, this would mean dividing the task among several people, thus complicating the administration and introducing inconsistencies. But the most interesting reason is that layout is not something that is simply added to text. Layout and wording interact. It therefore makes no sense for a program to generate a text without also specifying the layout features on which this text depends.

The aim of this paper is to illustrate some interactions between wording and layout, and to show how a NLG system can be organized in order to incorporate appropriate formatting specifications as the words are chosen. The work is based on the ICONOCLAST[1] project, in which we are developing an NLG system for generating patient information leaflets.

2 Interactions between wording and layout

2.1 Referring expressions

Many linguists have observed that pronouns are rarely used when their antecedents lie across a major boundary in the text (Hoffman, 1989). Within a section, or a box, or a footnote, an entity is almost always mentioned first by a description. Even a decision to break a paragraph into two may require some rewording. Consider the following paragraph, adapted slightly from a patient information leaflet (ABPI, 1997).

> Cyprostat contains cyproterone acetate. This substance is an antiandrogen. It blocks the actions of male sex hormones. It also reduces the amount of male sex hormones produced by the body.

[1]ICONOCLAST is supported by the UK Engineering and Physical Sciences Research Council (EPSRC) Grant L77102.

The pronoun *it* in the third sentence refers to cyproterone acetate, not to Cyprostat. Now, suppose that the author for some reason introduces a paragraph break after the second sentence. One might dispute whether this is a good idea, but if it were done, the pronoun in the third sentence should surely be replaced by a description; otherwise there is a danger that it will be interpreted as referring to Cyprostat.

> Cyprostat contains cyproterone acetate. This substance is an antiandrogen.
>
> Cyproterone acetate blocks the actions of male sex hormones. It also reduces the amount of male sex hormones produced by the body.

In general, some spans of a document are presented as distinct graphical units, marked perhaps as paragraphs, or bulleted items, or boxes, or footnotes. Such spans serve as units that the reader can find quickly when scanning, and for this reason there is a tendency for them to be self-contained, so that referring expressions can be resolved either from background knowledge or from antecedents within the unit. A program therefore cannot generate appropriate referring expressions without considering which spans in the text will be expressed as graphical units.

2.2 Lists

In technical writing, vertical lists are often employed to express a series of semantically homogeneous items. All British patient information leaflets contain a section that lists the situations in which it might be unsafe to take the medicine, and another section listing possible side-effects (ABPI, 1997); in either case, the general form of the instruction is 'Tell your doctor if any of the following conditions holds'. The antecedent of this conditional is a complex disjunction which could in principle be presented through normal running prose:

> Tell your doctor if you experience poor appetite, a sick feeling, mild abdominal pain, alterations in your sense of taste, or pain in your muscles or joints.

However, in most leaflets a vertical list is preferred, in which case the connective *or* is omitted, and another phrase (e.g. *any of the following symptoms*) is usually added so as to link the conditional to the list of symptoms. Presumably the authors think it is obvious from the context that the list expresses a disjunction rather than a conjunction; the reason for including the connective in running prose is that it is required for correct syntax. Since vertical lists distinguish the elements of the antecedent by other means (e.g. indenting and bullets), the connectives and punctuation marks that perform these functions in running prose can be omitted.

> Tell your doctor if you experience any of the following symptoms:
>
> - poor appetite
> - a sick feeling
> - mild abdominal pain
> - alterations in your sense of taste
> - pain in your muscles or joints

2.3 Headings

When a phrase is presented as a heading (e.g. in bold face on a separate line), it must be suitably worded. Headings are usually noun phrases rather than full sentences. They usually

do not begin with an article: thus *Overdose* is preferred to *The overdose* or *An overdose*, as in the following excerpt from the Lamisil leaflet (ABPI, 1997, slightly modified):

> **Overdose**
> If you accidentally take too much of your medicine, tell your doctor immediately, or go to your nearest Casualty Department.

If the information in the heading were instead incorporated into the body of the section, then obviously some rewording would be necessary.

> In case of an overdose, i.e. if you accidentally take too much of your medicine, tell your doctor immediately, or go to your nearest Casualty Department.

Note that the linguistic constraints on section or chapter headings apply also to the item headings in description lists. In patient information leaflets it is even quite common for minor headings to be added to the front of a short paragraph, followed by a colon or dash.

3 Abstract and concrete features

The examples we have cited show that *some* features of graphical layout interact with wording. However, others plainly do not. Consider the following alternatives to the 'Overdose' example:

> OVERDOSE
> If you accidentally take too much of your medicine, tell your doctor immediately, or go to your nearest Casualty Department.

> **Overdose:** If you accidentally take too much of your medicine, tell your doctor immediately, or go to your nearest Casualty Department.

The first marks the heading by small capitals rather than bold face. The second incorporates the heading into the block of wrapped text, separating it by a colon; this was in fact the layout used in the original leaflet. These differences seem purely cosmetic, requiring no rewording of the text.

In this example, the feature that interacts with wording is the distinction between 'heading' and 'body'. When generating the wording, the program must specify that this distinction will be expressed *somehow* in the graphical layout, but it need not stipulate how: this decision can be taken later, in the confidence that no change need be made in the wording. 'Heading' exemplifies what we will call *abstract* layout features; 'bold face' and 'small capitals' are examples of *concrete* layout features. The abstract layout features may interact with wording; the concrete layout features are cosmetic.

Two further examples of abstract layout features are 'paragraph' and 'emphasis'.

- Paragraphs can be expressed by various combinations of concrete features: two common methods are a new line with a tab, and a longer vertical separation with no tab. Some patient information leaflets use the curious convention of marking each paragraph within a section with a bullet, regardless of whether the paragraphs express a series of points in parallel form. These differences appear to be cosmetic.
- Emphasis may be expressed by bold face, italics, small capitals, large capitals, or underlining. The choice among these concrete layout features seems cosmetic. It is the abstract feature of emphasis that interacts with wording, e.g. by using stronger wording when emphasis is not expressed by layout:

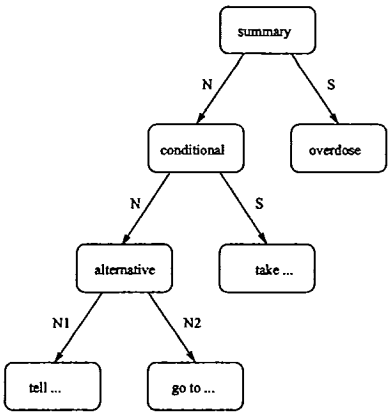

Figure 1: Semantic structure of a section

Never give your medicine to anyone else.
It is vitally important never to give your medicine to anyone else.

Among the abstract layout features, the most important are 'sentence', 'paragraph', 'section', 'itemized list', and the other *text-categories.* The notion of 'text-category' was introduced by Nunberg (1990) as part of a general distinction between *lexical-grammar*, which specifies how words may be combined syntactically, and *text-grammar*, which specifies at an abstract level how they will be presented in a written document. Thus a text-sentence is any span of the text that would normally begin with a capital letter and end with a full stop, while a lexical-sentence is any span that obeys the syntactic rules for a well-formed sentence, e.g. $S \rightarrow NP + VP$. Often these categories coincide, but sometimes they do not. A text-sentence may contain constituents that are lexical-sentences but not text-sentences:

Most people benefit from taking this medicine but a few people can be upset by it.

In some genres, such as thrillers, it is quite common to find lexical noun-phrases presented as text-sentences:

I opened the safe. Disaster. The money had gone.

Note that the danger of confusion between lexical-categories and text-categories arises only for sentences and lower units (clause, phrase). Paragraphs, sections, and itemized lists are obviously text-categories which play no role in syntax.

4 Generating documents with layout

The ICONOCLAST system, currently under development, allows an author to specify the content of a patient information leaflet, and then to generate many versions by varying a style profile. The style profile is presented as a set of 50-100 constraints concerning text structure, punctuation, terminology, syntax, and layout; a desired house style can be defined by turning these constraints on or off, and by setting any parameters that they contain (e.g. a limit on average paragraph size).

The input to the generator is a content model and a style profile. We model content by semantic networks that can be interpreted as Discourse Representation Structures (Power,

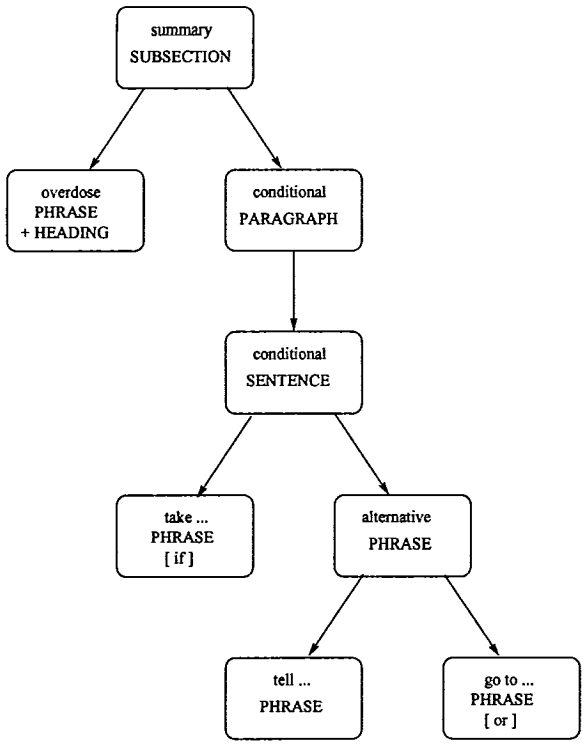

Figure 2: Semantic structure of a section

1999); the author edits the model by the WYSIWYM method (Power and Scott, 1998). Figure 1 shows part of the semantic network for the 'Overdose' example; to save space we have omitted the internal structure of propositions like 'you accidentally take too much of your medicine', focussing on the rhetorical relations (summary, conditional, alternative) which determine the large-scale structure of the text.

The generation process has three main phases.

Text Planning

The generator builds a hierarchical text plan (figure 2) which expresses the rhetorical relations in the semantic input without yet tackling elementary propositions. The text plan is a tree in which each node is assigned a text-category along with the other abstract layout features mentioned above (e.g. ±HEADING). While constructing the text plan, the generator must respect the relevant constraints in the style profile, along with some fixed constraints that the author is not allowed to edit (e.g. that paragraphs should be composed of sentences rather than vice-versa).

Tactical Realization

Having designed the overall structure of the text, the generator applies its grammar and lexicon in order to express the elementary propositions. It will be constrained in various ways by the output of the text planner, which dictates (a) the propositional content, (b) whether the proposition should incorporate a discourse connective (e.g. if, or), (c) the syntactic category of the unit expressing the whole proposition, and (d) whether a

major structural boundary has been crossed since a referent was last mentioned (this may require that a description will be used rather than a pronoun). In addition, the generator must respect any constraints on linguistic style imposed by the author, such as preferences over terminology or grammatical patterns.

Graphical Realization

During the final phase, the generator adds concrete layout features that realize abstract features like 'sentence' and 'heading'. Capital letters, full stops, and commas will be introduced for sentences and some phrases. Headings and paragraphs will normally be assigned to separate cells, separated by vertical space. Character formatting for text strings will be specified, so that for example headings are distinguished by bold face. Again, the assignment of these features must respect some fixed constraints as well as some constraints controlled by the author.

5 Conclusion

In essence, our approach is to separate the *abstract* layout features that interact with wording from the *concrete* layout features that are cosmetic. During text planning, abstract layout features such as text category are specified; these constrain the choice of words during tactical generation. The concrete layout features assigned during graphical realization serve to express the text categories and other abstract layout features, so that it is through the mediation of these abstract features that the detailed layout of the document is adapted to its wording.

References

ABPI (1997). Compendium of patient information leaflets. Association of British Pharmaceutical Industries.

Hoffman, T. (1989). Paragraphs and anaphora. *Journal of Pragmatics*, 13:239–250.

Nunberg, G. (1990). *The Linguistics of Punctuation.* CSLI/SRI International, Menlo Park CA.

Power, R. (1999). Controlling logical scope in text generation. In *Proceedings of the European Workshop on Natural Language Generation*, pages 1–9, Toulouse, France.

Power, R. and Scott, D. (1998). Multilingual authoring using feedback texts. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*, pages 1053–1059, Montreal, Canada.