Authoring on Demand: Natural Language Generation in Hypertext Documents

Robert Dale and Maria Milosavljevic

Microsoft Institute
Macquarie University
65 Epping Road
North Ryde NSW 2113

{rdale,mariam}@mpce.mq.edu.au

Abstract

Automatic natural language generation from underlying representations of information adds a new dimension to the electronic publishing world, particularly in the context of on-line information sources. In this paper, we provide an overview of our Peba-II natural language generation system, and focus on two key themes which take on special significance in the context of hypertext document generation: the notion of co-operative discourse planning, and the generation of text that is sensitive to the context.

1 Introduction

In this paper, we consider the use of natural language generation (NLG) techniques in the context of dynamically constructing hypertext documents (in fact, World Wide Web pages). Our interest is in building information provision systems which can present the same information in different ways at different times to different users, thus allowing the re-use of the underlying informational resources.

A great benefit of existing hypertext documents is that they allow user-directed exploration of the subject matter. However, many hypertext documents are simply linear documents broken into pieces which are then connected by links. By allowing the reader to explore the pieces of the resulting document in an order other than their original linear order, problems of discontinuity may arise. Ideally, the text presented should in some way be sensitive to where the user has been before. One solution to this problem is to dynamically generate the text from an underlying representation, rather than simply present canned pre-written text fragments. The techniques required are precisely those which form the core focus of research in natural language generation.

Proceedings of the First Australian Document Computing Conference, Melbourne, Australia, March 20–21 1996.

Another benefit of hypertext documents is that they effectively constitute a dialogue with the reader, who may decide at will whether or not to explore particular avenues. This characteristic has very interesting consequences for work in NLG. A major goal in NLG research is that of producing text that is tuned to the needs of a particular user. This requires a sophisticated approach to the problem of user modelling. The difficulty of providing an adequate characterisation of a user's existing knowledge poses serious problems for the task of text planning, which is concerned with determining what the overall content of a text should be. The hypertext medium alleviates some of these problems: decisions about what should be included can be made by the user herself.

In this paper, we explore both of these points, showing how the synergy of natural language generation and the hypertext medium can lead to dynamic documents which offer considerable potential as a means of information provision.

2 An Overview of the Peba-II Document Generation System

2.1 Natural Language Generation as a Goal

The general goal of research in NLG is the development of techniques that use some underlying representation of information to construct natural language texts. This is in contrast to work in natural language understanding (NLU), where the aim is to go from text to some underlying representation. For many researchers, the representations that serve as input to the generation process are the same as or similar to those that might be produced by a language understanding process, but this need not be the case.[1] One interesting difference of em-

[1] For example, it is probably true to say that the majority of work in natural language understanding focusses on the construction of representations that are close to formal logical languages; although such representations can play an important role in language generation, there are also NLG

Figure 1: A Web page generated by Peba-II

phasis between work in NLU and NLG is that, to date, most work in NLU has focussed on the sentence as a linguistic unit, being primarily concerned with deriving syntactic and semantic representations at that level. Work in NLG, on the other hand, often focusses on larger units of text: multi-sentence paragraphs and multi-paragraph documents. This difference of emphasis means that research in NLG attaches a greater importance to issues of discourse coherence and fluency, which are precisely the issues that we face in the context of hypertext information provision.

The motivations for research in NLG are many, but core to much recent work is the aim of producing different documents from one underlying source. These documents might vary in purpose, length, assumed age of the intended audience, or the natural language used; the underlying representation may often have been developed for some other purpose (for example, an expert system's knowledge base, or a CAD system's model of some artifact). Reports of recent research in the area can be found in Dale, Mellish and Zock [1990]; Paris, Swartout and Mann [1991]; Dale *et al* [1992]; and Horacek and Zock [1993].

Our current research in natural language generation is directed towards developing an intelligent on-line encyclopædia. Given some underlying knowledge base of information, we want to be able to dynamically produce natural language texts that are tailored to their context of use: what we generate today for one user might be quite different to that which was generated for the same or another user yesterday. In the 1980s, a key idea in document production allied to desktop publishing and laser printing was that of publishing on demand; in the 1990s, using NLG technologies opens up scope for a shift to what we might think of as authoring on demand.

2.2 The Peba-II Document Generation System

To explore this idea, we have constructed a simple prototype NLG system called Peba-II. This system takes an underlying knowledge base that contains information in our chosen domain (the animal kingdom) and produces from it textual descriptions,

systems which take as input simple collections of numerical data in tabular format as might be used to represent, for example, stock market data or meteorological data, and produce from these descriptive textual summaries.

Figure 2: The architecture of Peba-II

similar in content to encyclopædia entries. Each description is in the form of a World Wide Web page. Figure 1 shows an example output page which compares two animals.[2]

The overall architecture of the Peba-II system is shown in Figure 2. The text generator begins with some communicative goal provided by the user (effectively, a directive like 'describe the echidna' or 'compare the echidna and the platypus') and, taking account of the available linguistic resources and contextual constraints, produces a Web page that satisfies this goal. The linguistic realiser we use is Elhadad's [1992] FUF, combined with a small unification-based grammar of English we have developed for our domain. More details on these components can be found in Milosavljevic, Tulloch and Dale [1996].

A document renderer, which carries out the work that is required in order to realise the generated text in some medium, must ultimately play a role in any system which does more than generate disembodied texts [Dale 1992a]. In Peba-II, the document renderer is any Web browser such as Mosaic or Netscape, but could equally well be some other component which translates document structuring commands into a visible form. In the current system the document structuring commands used are a subset of HyperText Markup Language (HTML).

[2]A version of the system is available on the Web, and can be explored with any Web browser via URL `http://www.mpce.mq.edu.au/msi/peba.html`.

In operation, the user guides the system's processing by selecting hypertext tags which are used to indicate new discourse goals; each goal results in the generation of a Web page which contains a number of hypertext tags that correspond to a range of further discourse goals the user can choose to pose to the system; this results in a dynamic text planning enterprise where the user decides what information he or she would like to see on the next page generated.

2.3 The Underlying Knowledge Representation

The content of the generated texts is derived from a knowledge base of facts about animals, which in the current version of the system has been hand-constructed from an analysis of existing encyclopædia articles.

Our knowledge representation contains a Linnaean animal taxonomy, where the principal nodes are animal classes and whose arcs, represented using ako links, indicate subset and superset relationships between these classes. A fragment of the knowledge base hierarchy is shown graphically in Figure 3. Concepts in the knowledge base are paired with semantic and syntactic structures in a phrasal lexicon, which is used by the linguistic realiser to produce surface natural language expressions. An important aspect of our work here is that we make use of a hybrid knowledge representation whereby we can represent linguistically-oriented information at various 'degrees of cannedness': we only use complex representations where there is benefit to be gained from doing so, and make use of precompiled structures where appropriate. Other work in our group explores the automatic construction of such precompiled structures: see Tulloch and Dale [1995].

The animal hierarchy allows us to infer relationships between animals and animal classes and to describe these. It also permits inheritance of features so that, for example, we may assume that all the subtypes of the Mammal produce milk (unless they have some counter clause). The hierarchy currently forms the main backbone for hypertext generation, as will be seen later.

Each node in the hierarchy serves as a location off which properties of the entity in question can be hung. There are two types of properties in the knowledge base. The distinguishing-characteristic (dc) clauses single out the important property that indicates how one subtype of a node is distinguished from others (and thus justifies the taxonomic distinction); for example, from Figure 3, the characteristic that distinguishes the Monotreme from all other Mammals is that it lays eggs. The hasprop

Figure 3: A Fragment of the Knowledge Base

clauses enumerate the known properties of an entity: some examples are shown in Figure 4.

Our current knowledge base contains 1137 clauses describing 401 classes; another strand of our research efforts is looking at the automatic extraction of this kind of representation from existing texts.

3 Co-operative Text Planning

3.1 The Nature of the Problem

The focus of our work described here is in working out what the content of a text should be, and determining how that information should be organised into a coherent text. This is a major issue for NLG, and it is widely accepted that a model of the user plays a crucial role in determining both the content and organisation of a text. As mentioned earlier, constructing suitable user models turns out to be very difficult (see [Paris 1993]). In light of this, our current work attempts to take advantage of the fact that hypertext navigation of a document source has characteristics that are more akin to dialogues than monologues: the interactivity effectively permits the user to ask questions at any point and select from a variety of lines of inquiry, restricted only by the set of links made available by the document author. The result is an interesting symbiosis, where the user has the initiative and at the same time is constrained by the set of possibilities offered; the overall character is of a dialogue where the set of questions the user can ask at any given point are restricted. An additional benefit of this mode of interaction is that the system's capabilities are explicit, so the user is less likely to become frustrated than they would by a system which accepts free-form natural language queries but often responds with 'Query not understood' messages: Tennant's NLMenu database interface [Tennant *et al* 1983] used a similar idea to overcome problems of this kind.

This symbiosis has been used in the development of a number of other NLG projects which have married text generation with the hypertextual presentation of information: see, in particular, Moore [1995] and Reiter *et al* [1992, 1995]. Moore's PEA (Program Enhancement Advisor) system permits the user to ask followup questions by clicking on component words of earlier messages generated by the system as it critiques the user's Lisp code; Reiter *et al*'s IDAS system allows the user to interrogate the system for more information on specific concepts mentioned in on-line documentation. The Peba-II system is closest in spirit to the IDAS system, and shares some intellectual heritage (see Reiter and Dale [1992]); however, the fragments of text produced by Peba-II are generally larger, with each corresponding to an entire Web page, whereas in IDAS each contribution from the system is what in some hypertextual systems might be thought of as pop-up glossary items. The larger scale of Peba-II's discourse contributions allows us to focus on discourse-level issues of the kind discussed in Section 4.

3.2 Using Schemas to Generate Text

The range of texts we are interested in generating is sufficiently invariant that the schema-based approach to text generation, introduced by McKeown

```
(hasprop Echidna (linean-classification Family))
(distinguishing-characteristic Echidna Monotreme (body-covering sharp-spines))
(hasprop Echidna (nose long-snout))
(hasprop Echidna (social-living-status lives-by-itself))
(hasprop Echidna (diet eats-ants-termites-earthworms))
(hasprop Echidna (activity-time active-at-dusk-dawn))
(hasprop Echidna (colouring browny-black-coat-paler-coloured-spines))
(hasprop Echidna (lifespan lifespan-50-years-captivity))
```

Figure 4: A knowledge base fragment for the Echidna

[1985], makes most sense. Schemas essentially provide paragraph templates of pre-defined structure, content and order: for example, we can formulate a standard way to describe an animal which includes giving information about its name and taxonomy, distinguishing features, habitat, size and weight, followed perhaps by an example. In the general case such techniques are too rigid for fluent text production; however, some variation comes from the differing kinds of information available on any given animal, and the remaining elements of uniformity themselves have some value in an instructional context.

Each schema provides a set of ordering constraints over a pattern of RHETORICAL PREDICATES in such a way that the resulting text is fluent and coherent; each rhetorical predicate is effectively a representation of a speech act type, defined so as to provide an interface to the underlying knowledge representation. For our purposes, two schemas—which we call IDENTIFY and COMPARE AND CONTRAST—suffice. The COMPARE AND CONTRAST schema is represented by the following grammar rules, where each terminal symbol in the grammar corresponds to a rhetorical predicate:

(1) CompareAndContrast ⟶
  LinnaeanRelationship CompareProperties
CompareProperties ⟶
  CompareProperty CompareProperties
CompareProperties ⟶ ϕ

The LinnaeanRelationship rhetorical predicate describes how the animals are related according to the Linnaean animal taxonomy. This relationship is determined by traversing the hierarchy upwards from each animal until the lowest common ancestor is found.

We use an underlying corpus-derived categorisation of properties into a hierarchy to permit appropriate comparisons to be drawn; this allows us to determine that, for example, height and length are both measurements of size and so can be usefully mentioned together. Using this hierarchy of properties, the CompareProperty rhetorical predicate searches for related properties for the selected animals.

Figure 1 shows a Web page created by the COMPARE AND CONTRAST schema. The underlined entities are clickable hypertext which indicate new discourse goals for the text generation system. Note here another benefit of using natural language generation techniques: by representing separately the information about 100 animals, we have the resources available to dynamically generate 4950 different comparisons. This is clearly more space-efficient than storing even a high-probability subset of these documents in canned form. It also makes it much easier to deal with updates to the knowledge represented; changes need only be made in one place, and all subsequent documents generated will reflect this change.

3.3 Discourse Plans and Sentence Plans

The effect of instantiating a schema with respect to some portion of the knowledge base is a discourse plan, which specifies the propositional content of the speech acts that make up the text to be generated. Figure 5 shows an example discourse plan. Each speech act specification results in the generation of a surface sentence.

Each speech act may contain references to conceptual entities which are marked as hypertext jumping-off points. In the current system, only entities corresponding to nodes in the animal taxonomy are marked in this way, although in principle any concepts introduced in the text (such as, for example, *spines* in the description of the Echidna in Figure 1) could be so marked. It is precisely this capability that allows what amounts to shared discourse planning: the underlying system uses schemas to determine what information should be included, but the user can choose to interrogate further at any point. Reiter *et al*'s [1995] IDAS system takes advantage of this observation to limit the amount of information presented at any one time.

```
((schema-type identify)
 (constituents
      ~(((speech-act-type name-entity)
         (content ((primary-name ((cat np) (sem echidna) (name-type name)))
                   (secondary-name
                         ((cat np) (sem echidna) (name-type common-name)))
                   (supertype ((cat np) (sem monotreme) (name-type name)))
                   (relationship ((sem is-a-type-of)))
                   (distinguishing-characteristic
                         ((cat vp) (sem sharp-spines))))))
        ((speech-act-type list-subtypes)
         (content ((head ((cat np) (sem echidna) (name-type name)))
                   (arguments
                    ~(((cat np) (sem short-beaked-echidna) (name-type name))
                      ((cat np) (sem long-beaked-echidna)
                                (name-type name)))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp)
                              (sem lifespan-50-years-captivity))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp)
                          (sem browny-black-coat-paler-coloured-spines))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp) (sem active-at-dusk-dawn))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp)
                              (sem eats-ants-termites-earthworms))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp) (sem lives-by-itself))))))
        ((speech-act-type describe-property)
         (content ((name ((cat np) (sem echidna) (name-type name)))
                   (property ((cat vp) (sem long-snout))))))))
```

Figure 5: The discourse plan for a Compare and Contrast page

4 Discourse History in Non-linear Documents

4.1 The Role of Discourse History in Hypertext Documents

It is widely acknowledged that the interpretation of a piece of text depends on what has gone before. The simplest example of this is the use of anaphoric referring expressions such as pronouns, which are generally used to refer to entities mentioned in the previous discourse. Computational modelling of linguistic phenomena beyond the scope of a single sentence thus requires that we make use of some model of the discourse history; Dale [1992b] provides a detailed analysis of these problems from the point of view of language generation.

In linear documents, we are all familiar with the pitfalls of just 'dipping in' to a book at a random location: books (apart from certain kinds of reference works, of course) are usually written with linear traversal in mind, and so things can become confusing if we jump around the text in a relatively undisciplined manner. There is a clear warning here to anyone who would construct a hypertext document by simply breaking a linear text into fragments.

In reality, of course, a hypertext author can limit problems of this kind by controlling the links available at any given point. This is effectively what we do in Peba-II by only making the linguistic realisations of specific concept nodes be hypertext links. This is one means of ensuring that the reader only reaches texts that make sense in context, by imposing a global coherence or connectivity on the thread of conversation. There are interesting parallels here with the notion of CENTERING found in work on anaphora resolution [Grosz, Joshi and Weinstein 1983]: in that work, each sentence is seen as having a backward looking center (effectively, the entity that was talked about in the previous sentence) and a number of forward looking centers (entities mentioned in the current sentence that might be picked up on for subsequent elaboration in the next sentence). In the context of our Web page generation, each page, or utterance, has a backward looking center and a number of forward looking centers: the only difference is the domain of these concepts is the entire page of text rather than a single sentence.

4.2 What You See Depends on Where You Came From

There are a number of ways in which we can vary the text the reader sees based upon other text they have seen. Here we examine more closely the effects on natural language generation of taking account of the information seen by the reader immediately prior to the new page or utterance being constructed.

It is easiest to see what happens here with reference to the fragment of the taxonomy shown in Figure 3. Because our IDENTIFY schema always relates a class to its superordinate class and its subordinates, any node in the taxonomy may be reached from one of two directions: either from the node immediately underneath, or from the node above. Different texts can be produced in each case.

Suppose we are at the beginning of a session with the system where no previous nodes have been visited, and the user requests a description of the marsupial class; we might then generate the following text:

(2) The Marsupial is a type of Mammal that carries its young in a pouch. The Marsupial has the following subtypes ...

However, suppose the user reaches the marsupials node from the mammals node, as would be the case after reading the following text and then clicking on the marsupials link:

(3) There are three kinds of mammals:

- the monotremes
- the placental mammals
- the marsupials

This would give rise to the text in example (4):

(4) The Marsupial differs from other Mammals in that it carries its young in a pouch. It has the following subtypes ...

On the other hand, suppose we reach the marsupial node from viewing a description of the kangaroo class; ie, after reading the following text:

(5) The kangaroo is a kind of marsupial which has a powerful tail and back legs.

Here, if we click on marsupial, then we reach the marsupials node from below, with the resulting text as shown in example (6).

(6) Apart from the Kangaroo, the class of Marsupials also contains the following subtypes ...

By varying the way in which the information is presented in this way, we can generate more fluent texts. In each case, the way in which the queried class is introduced is determined by how best this fits into the ongoing discourse.

We have only begun to explore the potential here for varying the text to take account of the discourse history: in the example just given, this is analogous to modifying the introductory section of a document for different audiences but leaving the body of the text substantially the same. However, some of our current work focusses on how particular concepts can be introduced by making reference to other concepts the user is assumed to know about, either by virtue of some independently constructed user model or because we know what previous concepts they have visited in a session with the system. This is analogous to taking account of notions of global discourse focus [Grosz 1977], taken here to correspond to the set of concepts assumed known to the reader on the basis of the entire preceding discourse; this permits a rather more radical variation in the texts created, and is the subject of some of our ongoing research [Milosavljevic 1996].

5 Conclusions and Future Work

In this paper, we have described our Peba-II document generation system, and discussed two characteristics of generating natural language in a hypertext environment. We have shown how the interactive nature of hypertext benefits NLG by removing some of the difficulties involved in text planning and user modelling; and we have shown how the dynamic generation of natural language can help overcome some of the pitfalls of exploring information in an unpredicted order.

In future work, we intend to explore further how the use of NLG techniques can complement hypertext presentation. In particular, we are interested in exploring the introduction of new concepts by relating them to concepts already known to the reader; this increases the role of the discourse history in determining what information a text should contain, but also frees us from the current limitations imposed on navigational possibilies made available to the reader by the use of the taxonomic hierarchy as a backbone.

On a more general and philosophical level, these experiments have caused us to reconceptualise what we have been doing. We started out exploring the idea of generating hypertext documents, but this has led us to view the activity as sequential utterance generation in an ongoing dialogue, where each utterance is a Web page. In this context, a discourse is the full interleaved sequence of Web pages and user queries (ie mouse clicks on links) that make up a session with the system.

This leads us to wonder about what we can validly call a document in such an interactive environment. Perhaps the notion of a document is itself too static and outdated to be of value here, and notions from sociolinguistics and conversational analysis — terms like turn and exchange — might be more fruitful metaphors. After all, even a static document such as a book can be thought of as just one very large contribution to an ongoing conversation with the audience.

Acknowledgements

We would like to thank Mark Lauer, Adrian Tulloch and Mike Johnson for their useful comments in the development of this work.

References

Robert Dale [1992a] Visible Language: Multimodal Constraints in Information Presentation. In Robert Dale, Eduard Hovy, Dietmar Rösner and Oliviero Stock (eds) *Aspects of Automated Natural Language Generation*, Springer-Verlag.

Robert Dale [1992b] *Generating Referring Expressions.* MIT Press.

Robert Dale, Chris Mellish and Michael Zock (eds) [1990] *Current Research in Natural Language Generation.* Academic Press.

Robert Dale, Eduard Hovy, Dietmar Rösner and Oliviero Stock (eds) [1992] *Aspects of Automated Natural Language Generation.* Springer-Verlag.

Michael Elhadad [1992] Using Argumentation to Control Lexical Choice: A Functional Unification Implementation. PhD Thesis, Columbia University.

Barbara Grosz [1977] The Representation and Use of Focus in Dialog. Technical Note 151, SRI International, Menlo Park, California.

Barbara Grosz, Aravind Joshi and Scott Weinstein [1983] Providing a Unified Account of Definite Noun Phrases in Discourse. Pages 44–50 in *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, MIT, 15–17 June 1983.

Helmut Horacek and Michael Zock (eds) [1993] *New Concepts in Natural Language Generation.* Academic Press.

Kathleen McKeown [1985] *Text Generation.* Cambridge University Press.

Maria Milosavljevic [1996] Introducing New Concepts via Comparison: A New Look at User Modelling in Text Generation. In *Proceedings of the Fifth International Conference on User Modelling—Doctoral Consortium*, Hawaii, January 1996.

Maria Milosavljevic, Adrian Tulloch and Robert Dale [1996] Text Generation in a Dynamic Hypertext Environment. In *Proceedings of the Nineteenth Australasian Computer Science Conference*, Melbourne, March 1996.

Johanna Moore [1995] *Participating in Explanatory Dialogues.* MIT Press.

Cecile Paris [1993] *User Modelling in Text Generation.* Pinter Publishers.

Cecile Paris, William Swartout and William Mann [1991] *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Kluwer Academic Publishers.

Ehud Reiter and Robert Dale [1992] A Fast Algorithm for the Generation of Referring Expressions. In *Proceedings of Coling–92*, Nantes, France, August 1992.

Ehud Reiter, Chris Mellish and John Levine [1992] Automatic Generation of On-line Documentation in the IDAS Project. In *Proceedings of the Third Conference on Applied Natural Language Processing*, Trento, Italy, March–April 1992.

Ehud Reiter, Chris Mellish and John Levine [1995] Automatic Generation of Technical Documentation. *Applied Artificial Intelligence*, Vol 9(2), pp 259–287.

Harry Tennant, Kenneth Ross, Richard Saenz, Craig Thompson and James Miller [1983] Menu-based Natural Language Understanding. Pages 151–158 in *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, MIT, 15–17 June 1983.

Adrian Tulloch and Robert Dale [1995] Speeding up Linguistic Realisation by Caching Previous Results. In *Poster Proceedings of the Eighth Australian Joint Artificial Intelligence Conference*, Canberra, November 1995.