A HYBRID APPROACH TO THE AUTOMATIC PLANNING OF TEXTUAL STRUCTURES

Gang Zhu and Nigel Shadbolt

A.I. Group, Psychology Department, Nottingham University, Nottingham, U.K.

1. Introduction

Over the last decade, the research goals in natural language generation have shifted from the generation of isolated sentences to the production of coherent multi-sentence paragraphs. Two major aspects of the generation process have been focused on: deciding 'what to say' (the strategic level) and deciding 'how to say it' (the tactical level).

In 1985, McKeown designed one of the first systems to produce paragraphs using so-called schemata to describe conventional text structures in terms of patterns. Schemata are used to determine the content and order of the clauses in paragraphs (McKeown, 1985). However, these structures have a major limitation (Moore and Paris, 1988): schemata do not contain a description of the intentional and rhetorical role that each part of the paragraph plays with respect to the whole paragraph.

In 1988, Hovy first employed RST (Rhetorical Structure Theory) relations, which state the relationships between individual elements of a text, to control the construction of texts (Hovy, 1988). In developing this RST-based method, Hovy has discovered that RST relations are a powerful tool for planning paragraphs. They support reasoning about the intentions of writers and readers in a very natural way. Planning with rhetorical relations affords more flexibility than schemata. This method of planning paragraphs builds a tree structure that represents the internal organisation and rhetorical dependencies between clauses in a text. But there is a cost: it is more difficult to assemble an RST paragraph tree from a set of independent relations than it is to instantiate and traverse a schema (Hovy, 1991).

In 1992, Hovy et. al. described a new text planner (Hovy et. al., 1992) that identifies the distinct types of knowledge necessary to generate coherent discourse in a text generation system. These knowledge resources are integrated under a planning process that draws from appropriate resources whatever knowledge is needed to construct a text. Though Hovy et. al. do not claim to have identified all the knowledge sources required to produce coherent discourse, their planner sets a trend for applying multi-knowledge resources for more complete and flexible planning of text.

So far, planning techniques have developed from the direct application of schemata toward the wider implementation of multi-knowledge resources and diverse planning architectures (McKeown, 1985; Paris, 1987; Hovy, 1988; Moore, 1989; McKeown et al., 1990; Suthers, 1991; Hovy et. al., 1992). When these planning mechanisms are implemented in a working system, efficiency is still an important factor in developing a workable model. One of the problems in generation is that of designing a planning architecture that can achieve a good balance between the efficiency of the schema-based approach and the flexibility of the RST-based one. This paper presents such a hybrid architecture.

2. A Hybrid Approach

Both schema-based and RST-based planning paradigms have advantages and disadvantages. A hybrid of the two approaches that preserves their best aspects — the efficiency of the schema-based paradigm and the flexibility of the RST-based one — would clearly be useful. What are the possibilities for such a hybrid approach?

Though the two paradigms seem very different, the fact is that a close relationship exists between them. Schemata are nothing other than stereotypically occurring collections of plans, whereas the plans and their plan elements are simply the elementary building blocks of schemata (Mann, 1987). Schemata can be viewed as the result of a process where the plans for all of the steps in the process have been compiled into a single structure (Moore and Swartout, 1991). Schemata can be used for planning relatively invariant aspects of text content and structure. RST-based plans can cope with less predicable and more volatile. Both planning paradigms can be implemented if they are properly represented and manipulated in a hybrid architecture.

Two features are of importance in this hybrid approach: (1) different planning mechanisms are required to deal with different textual phenomena and (2) explicit use of multi-knowledge resources indispensable to these mechanisms.

In knowledge resources, there are two types of prescriptive knowledge: domain-dependent and the domain-independent knowledge. Both domain-de-

13. The Five Star support package,
14. costing 35% of the licence fee
15. (with a minimum of £500),
16. also includes free, automatic upgrades
17. (i.e. major releases)
18. to the products
19. that you have registered,
20. preferential rates for adding extra copies of those or any other products and memberships of the new LPA User group
21. Both support packages include discounts on any training or consultancy
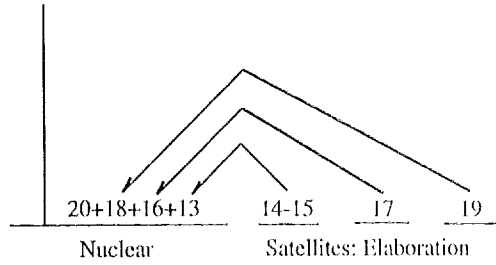22. ordered during the support period.

Fig. 1: A sample RST analysis

pendent and domain-independent knowledge resources will consist of intentional and rhetorical operators with associated hierarchical networks.

In the hybrid planning mechanism advocated here, top-down hierarchical expansion is used as the basic planning mechanism. It retrieves as much information as possible from relevant knowledge resources. In general it does this in a domain-dependent-to-domain-independent order. This order reflects the idea that an efficient planning mechanism should seek to exploit, whenever possible, stereotypical domain-dependent knowledge resources. This top-down planning mechanism is combined with other heuristic mechanisms such as an augmented transition network traversal, constructive critics and focus modules. This approach makes use of different knowledge resources and planning mechanisms and is capable of handling a number of different textual phenomena.

3. A Prototype

A prototype has been designed to demonstrate this hybrid approach to the problem of planning textual structures. We will first describe how the text data were collected and analysed. Based on this data, we will then discuss the knowledge resources that were identified as important as well as how they are represented. Finally we indicate how the planning mechanisms are implemented.

3.1. Test Data

The test data are a selection of English sales letters. These letters are relatively formalised, in that some paragraphs are fixed while others are more varied as to whether they appear and where they appear. The letters were written for a restricted readership on a specific subject — namely, certain computer software products.

The textual analysis has been carried out according to RST, although several modifications have had to be made. An example of part of a RST analysis is given in Fig. 1.

Although RST provides a framework for describing rhetorical relations among parts of a text, it lacks an explicit representation of the communicative intentions underlying the generation of coherent multisentential text. In order to construct a hybridised processor for various knowledge bases and planning mechanisms, we cannot implement RST directly with its rhetorical relations, but have to develop additional intentional relations. RST has to be supplemented with a richer intentional context.

3.2. Knowledge Resources

To plan our sales letters, we need to develop distinctive domain-dependent and domain-independent knowledge resources and their associated processing mechanisms in a planning system.

Each resource represents both domain-dependent information and domain-independent information. Hierarchical networks describe relationships among the contents of our knowledge resources.

In this section, we present the main knowledge resources that we have so far identified, namely: intentional operators, rhetorical operators, and networks over them.

3.2.1. Intentional Operators

Intentional operators are organised around the intentions of the writer, and their decompositions are used to select relevant rhetorical operators or appropriate speech acts as defined by Allen (1987). An intentional operator is represented using the formal theory of rational interaction developed by Cohen, Levesque, and Perraut (1985). Each operator has a goal, prerequisites, constraints, subgoals, and a type. The goal will be brought about by an application of the operator. The subgoals must be achieved for subsequent application of the operator. The prerequisites are conditions which must be satisfied, and constraints are conditions which can be ignored if there is no other intentional operator which has the desired goal. The type in each operator is either domain-dependent or

335

domain-independent. The criteria for the division between the domain-dependent and the domain-independent operators is based on the stereotypic patterns of our analysed texts. For example, Fig. 2 represents a domain-dependent intentional operator, Persuade. In our system, this operator may be instantiated as an attempt by an agent X to persuade a client Y to take an Action such as buying the agent's Products. This is achieved by making the client aware of the products and increasing his desire to take the action of buying the product. The prerequisites indicate that both the agent X and the client Y mutually believe that Information is about Products, the agent believes Information, and the client does not know it. These prerequisites must be satisfied within the existing knowledge resources before the intentional operator can be applied. The constraints, in this case that the client Y is not competent to fulfil Action, need to be satisfied at this stage of processing. When the constraints happened not to be satisfied within the existing knowledge resources, the constraints are then set as a new subgoal for later expansion.

GOAL: persuade(X, Y, Action)
PREREQUISITES:
bmb(X, Y, is(Information, Products))
bel(X, Information), not(know(Y, Information))
CONSTRAINTS: not(competent(Y, Action))
SUBGOALS: bmb(X, Y, and(bel(Y, Products), increase_desire(Y, Action))
TYPE: domain-independent

Fig. 2: An intentional operator: Persuade

3.2.2. Rhetorical Operators

Rhetorical operators are associated with intentional operators. This association reflects the fact that there are certain rhetorical means of achieving particular intentional goals. Rhetorical operators consist of seven components: Prerequisites, Constraints, Effects, Nuclear, Satellite, Order and Type. As with our intentional operators the prerequisites must always be satisfied. Constraints may be ignored but if they are processed they have the same potential as constraints in intentional operators -- they may become new goals for the system. Rhetorical operators as expected to have clear effects on intended recipients. Our rhetorical operators also possess the important constituents of a nuclear and satellite. They concern how the goals expressed in the calling intentional operators are to be achieved -- the actions to be carried out. There are two types of rhetorical operators—domain-dependent and domain-independent:

Domain-independent rhetorical operators are general rhetorical operators applicable across a wide range of types of texts. There are about thirty of them described to date (Mann and Thompson 1987). Planning with these operators affords more flexibility than schemata, because individual operators typically control less of a paragraph than schemata do;

Domain-dependent rhetorical operators are derived from our RST analysis of our task-oriented data. Having analysed our sales letters we have identified those rhetorical operators that seem particular to such computer product sales texts. Often they are rather schematic in that one can expect certain material to be expressed in particular ways at certain parts in the text.

3.2.3. Intentional and Rhetorical Networks

The intentional network is a hierarchical structure that embodies a preferred control structure for the use of our intentional operators. The intentional network can be used for giving possible development of communicative goal(s) with heuristic ordering for an efficient schema-based approach.

The rhetorical network is derived from several main sources: the relations defined in RST (Mann and Thompson 1989), which were extended in Hovy's taxonomization of relations (Hovy et. al. 1992), and others as determined by our sales-letter domain. This rhetorical network operates together with the other knowledge resources, by posting the hierarchical patterns of intentional operator(s), selecting relevant speech act(s), or specifying aspects of grammatical realisation.

3.3. Planning Mechanisms

A text planner, in the form of a heuristic planning process adopted from the layered architecture JAM (Carletta 1992) and a top-down hierarchical expansion system based on NOAH (Sacerdoti 1977), has been implemented to plan coherent paragraphs which achieve a goal. The goal is configured with initial states designed to affect a reader in a specified way.

During the main planning process, top-down hierarchical planning takes place. This occurs when intentional operators are expanded into a network of subgoal(s), or rhetorical operators are expanded into a network of actions. Planning is also involved when unsatisfied constraints become new subgoals. There may be several alternative expansions to be explored. At this point, the organisation of the plan expressed by one or more structure trees may have to be criticised to account for interactions between parts of what were previously unanalysed subgoals and actions. If there exist a group of structure trees, these trees have to be focused through selective heuristics.
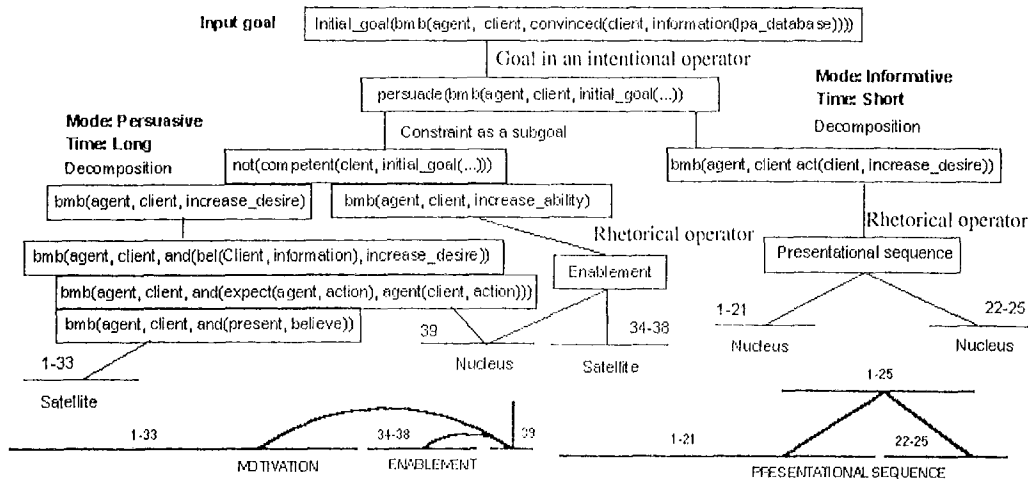
Fig. 3: A simplified top-level planning process for two alternative textual structures

These heuristics prefer structures with less subgoals remaining or lower cost estimates in the knowledge hierarchical networks. We call these critic processes heuristic ordering mechanisms.

For example, Fig. 3 shows a simplified top-level planning process for two alternative textual structures. The initial goal is that the writer or agent wishes to convince the client about information concerning on LPA database products. The two alternative structures of Fig. 3 represent two different plans that our system can generate so as to achieve the initial goal. The two plans vary in terms of whether the text is lengthy and persuasive, else short and informative. The persuasive, lengthy setting results in an operator being selected to increase the client's desire to buy the products. But a constraint of the original persuade operator is expanded. The operator attempts to increase the client's ability to take advantage of his strengthened desire to buy the products. This will result in text that attempts to produce

a means of enablement to increase the ability to satisfy the desire. Motivation and Enablement are used to produce a partial textual structure on the left side of Fig.3. Otherwise, when an Informative mode and a Short time setting are required, the system selects an intentional operator with a rhetorical operator to fulfil its initial goal as shown on the right side of Fig. 3. This is a simplified presentation of informing material about the products.

The output for the hybrid planner is a single structure tree, with speech acts associated with each of the terminal nodes. The terminal nodes specify propositions discharging those speech acts. This information is chosen so that, with minor supplementation, it is sufficient to specify sentences to be generated by a functional grammar (see Fig. 4)

During the process of developing the hybrid planning prototype, we have found that it possesses the following advantage. Heuristic strategies can be imple-

2. [[[[[bmb(agent,client,and(agent is pleased, present(agent,agent is pleased)))],
* LPA is pleased
3. [[[[bmb(agent,client,and(announce(agent,window_series), bel(client,window_series)))],
* to announce its new Windows series,
4. [[[[bmb(agent,client,and(inform_object(agent,machine_386 / 486),bel(client,machine_386 / 486)))],
* a range of software tools for 386 and 486 machines
5.
[bmb(agent,client,and(inform_attribute(agent,windows_3_in_enhanced_mode), bel(client,windows_3_in_enhanced_mode)))]]]]]]]],
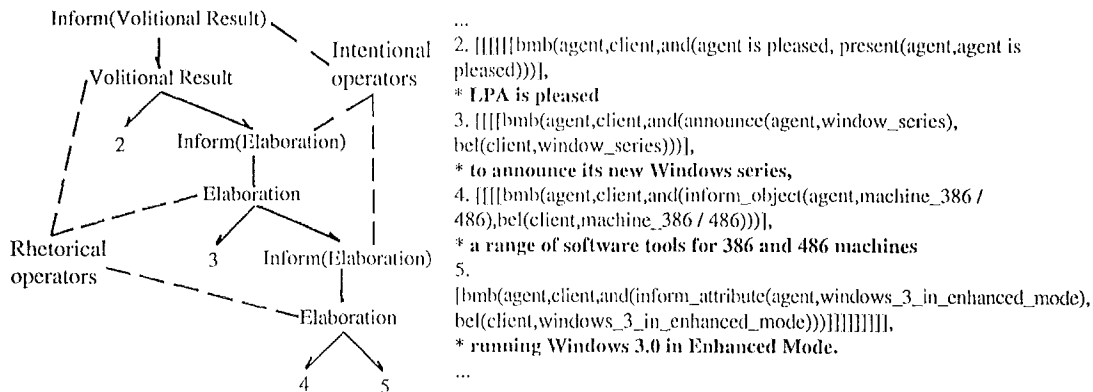* running Windows 3.0 in Enhanced Mode.
...

Fig. 4: A sample partial output tree

mented within a non-linear hierarchical planning procedure and multi-knowledge resources can be employed selectively at each level of abstraction. Its top-down hierarchical expansion process provides an efficient non-linear planning mechanism. Its heuristic strategy flexibly chooses not to expend all of the effort needed to employ various resources unless it is absolutely necessary.

4. Conclusion

This paper has presented a hybrid approach to the planning of textual structures. It is based on the idea that a variety of explicit knowledge resources and planning mechanisms are needed for an efficient but flexible text planner. By describing a hybrid planning prototype, it identifies various knowledge resources required in the domain of business letters. It suggests associated planning techniques for manipulating intentional and rhetorical information. Since the research is still in progress, this paper cannot claim to have identified all the necessary knowledge resources and requisite planning mechanisms. Consequently, certain problems, such as how to evaluate various planning critics in detail, remain unsolved. The next stage of the research is to capture richer knowledge in the domain and further develop the critic modules and their controlling mechanisms. Nevertheless we feel that the system as it stands represents a linguistically motivated and coherent computational architecture for the generation of text. The generated text is, moreover, rhetorically compelling given the intentional goals of the originator.

References

Allen, J. 1987. Natural Language Understanding, The Benjamin/Cummings Publishing Company, Inc..

Carletta, J. 1992. *Risk-taking and Recovery in Task-Oriented Dialogue*, PhD thesis, University of Edinburgh

Cohen, P. R. and Levesque, H. 1985. Speech acts and nationality. *In* Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics.

Hovy, E. H. 1988. Planning coherent multisentential text. *In* Proceedings of the twenty-sixth Annual Meeting of the Association for Computational Linguistics, State University of New York, Buffalo, New York.

Hovy, E. H. 1990. Unsolved issues in paragraph planning. *In* R. Dale, C. Mellish and M. Zock *Eds.*, *Current Research in Natural Language Generation*. Academic Press Limited, London.

Hovy, E.H. 1991. Approach to the planning of coherent text. *In* C.L. Paris, W.R. Swartout and W.C. Mann *Eds.*, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Kluwer Academic Publishers, USA.

Hovy, E., Lavid, J., Maier, E., Mittal, V. and Paris, C. 1992. Employing knowledge resources in a new text planner architecture. *In* R. Dale, E. Hovy, D. Rosner and O. Stock *Eds.*, *Aspects of Automated Natural Language Generation*. Springer-Verlag Berlin Heidelberg.

Mann, W.C. 1987. Text Generation: the Problem of Text Structure. Technical Report No. RS-87-181, USC/Information Science Institute, Marina Del Rey, CA.

Mann, W. C., Matthiessen, C. and Thompson, S. A. 1989. Rhetorical Structure Theory for text analysis, USC/Information Sciences Institute, Technical Report ISI/RR-89-242.

McKeown, K. R. 1985. *Text Generation : Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press, Cambridge, England.

McKeown, K. R; Elhadad, M.; Fukumoto, Y.; Lim, J.; Lombardi, C.; Robin, J. and Smaadja, F. 1990. Natural language generation in COMET. *In* R. Dale; c. Mellish and M. Zock *Eds.*, *Current Research in Natural Language Generation*. Academic Press Limited, London

Moore, J. D. 1989. A reactive approach to explanation in expert and advice giving systems. PhD. dissertation, University of California, Los Angeles, CA.

Moore, J.D. and Paris, C.L. 1988. Constructing coherent text using rhetorical relations. *In* Proceedings of the National Conference on Artificial Intelligence, Moston, MA.

Sacerdoti, E. D. 1977. A Structure for Plans and Behaviours. New York: North Holland.

Suthers, D. S. 1991. A task-appropriate hybrid architecture for explanation. Computational Intelligence, 7(4).