From local to global coherence: A bottom-up approach to text planning

Daniel Marcu
Department of Computer Science
University of Toronto
Toronto, Ontario
Canada M5S 3G4
marcu@cs.toronto.edu

Abstract

We present a new, data-driven approach to text planning, which can be used not only to map full knowledge pools into natural language texts, but also to generate texts that satisfy multiple, high-level communicative goals. The approach explains how global coherence can be achieved by exploiting the local coherence constraints of rhetorical relations. The local constraints were derived from a corpus analysis.

Motivation[1]

All current flexible approaches to text planning that assume that the abstract structure of text is a tree-like structure are, esentially, top-down approaches. Some of them define plan operators and exploit hierarchical planning techniques (Hovy 1993; Moore and Paris 1993; Moore and Swartout 1991; Cawsey 1991; Maybury 1992) and partial-order planning techniques (Young and Moore 1994). Others assume that plans are hierarchically organized sets of frames that can be derived through a top-down expansion process (Nirenburg *et al.* 1989; Meteer 1992). And the recursive application of schemata (McKeown 1985) can be thought of as a top-down expansion process as well.

One of the major strengths of all these approaches is that, given a high-level communicative goal, they can interleave the task of text planning and content selection, and produce different texts for different knowledge bases and users (McKeown 1985; Paris 1991; McCoy and Cheng 1991; Moore and Swartout 1991). Unfortunately, this strength is also a major weakness, because top-down and schema-based approaches are inadequate when the high-level communicative goal boils down to "tell everything that is in this knowledge base or everything that is in this chosen subset". The reason for this inadequacy is that these approaches cannot ensure that *all* the knowledge that makes up a knowledge pool will be eventually mapped into the leaves of the resulting text plan: after building a partial text plan, which encodes a certain amount of the information found in the initial knowledge pool, it is highly possible that the information that is still unrealized will satisfy none of the active communicative goals. In fact, because the plan construction is plan-operator- or schema-step-driven, top-down approaches cannot even predict what amount of the initial knowledge pool will be mapped into text when a certain communicative goal is chosen. The only way to find a text plan that is maximal with respect to the amount of knowledge that is mapped into text is to quantify over all possible high-level communicative goals and over all plans that can be built starting from them, but this is unreasonable.

Given that most natural language generation (NLG) systems employ a pipeline architecture in which content determination and text planning are treated as separate processes (Reiter 1994), we believe that it is critical to provide a flexible solution to the problem of mapping a full knowledge base (or any of its chosen subsets) into text. Previous research in text planning has addressed this issue only for text genres in which the ordering of sentences is very rigid (geographical descriptions (Carbonell and Collins 1973), stories (Schank and Abelson 1977), and fables (Meehan 1977)), has assumed that text plans can be assimilated with linear sequences of textual units (Mann and Moore 1981; Zukerman and McConachy 1993), or has employed very restricted sets of rhetorical relations (Zukerman and McConachy 1993). Unfortunately, the linear structure of text plans is not sophisticated enough for managing satisfactorily a whole collection of linguistic phenomena such as focus, reference, and intentions, which are characterized adequately by tree-like text plans (Hovy 1993; Moore and Paris 1993; Moore and Swartout 1991; Cawsey 1991; Paris 1991; McCoy and Cheng 1991).

In this paper we provide a *bottom-up, data-driven solution for the text planning problem* that implements the full collection of rhetorical relations that was proposed by Mann and Thompson (1988) and that accommodates the hierarchical structure of discourse. The algorithms that we propose not only map a knowledge pool into text plans whose leaves subsume all the information given in the knowledge pool, but can also ensure that the resulting plans satisfy multiple high-level communicative goals.

Foundations of our approach

Introduction

Let us assume that we are given the task of constructing a text plan whose leaves subsume all the information given in a knowledge base (KB). For simplicity, we assume that the KB is represented as a set of semantic units

$U = \{u_1, u_2, \ldots, u_n\}$. We also assume that rhetorical relations of the kind described by Mann and Thompson (1988) hold between pairs of semantic units in U. These rhetorical relations can be derived from the KB structure, from the definitions of a library of plan operators, or can be given as input by the creator of the KB. For example, if the semantic units are stored in a description-logic-based KB such as LOOM or CLASSIC, one can derive some rhetorical relations by inspecting the types of links and paths between every pair of semantic units. When the KB consists of a set of frames with clearly defined semantics, such as those produced by systems developed for information extraction tasks, one can use the underlying semantics of frames to derive rhetorical relations between the information encoded in different slots. For less structured KBs, one can use the collection of libraries of plan operators that were developed by researchers in hierarchical planning (Hovy 1993; Moore and Paris 1993; Meteer 1992) and derive the set of rhetorical relations that hold between every pair of semantic units. For very rich KBs, such as that described by DiMarco et al. (1995) and Hovy and Wanner (1996), one can simply extract these relations directly, because they are explicitly represented.

Each of the alternatives described above has been already discussed in the literature to a greater or lesser extent. Therefore, for the purpose of this paper, we will simply assume that the input for a text planner is a set U of semantic units and the set R of rhetorical relations that hold between every pair of units in U. Note that there are no constraints with respect to the number of rhetorical relations that may hold between two semantic units: on one hand, when two units are not related, no rhetorical relation holds between them; on the other hand, depending on the communicative goal that one wants to emphasize, more than one relation may hold between two units (Mann and Thompson 1988; Moore and Pollack 1992). In the latter case, we assume that R lists all possible relations.

For example, the KB in (1) contains four semantic units among which four rhetorical relations hold (2).

$$(1) \quad U_1 = \left\{ \begin{array}{l} a_1 = \text{“Insulin-dependent diabetes is} \\ \quad \text{the less common type of diabetes.”} \\ b_1 = \text{“The pancreas, a gland found} \\ \quad \text{behind the stomach, normally makes} \\ \quad \text{insulin.”} \\ c_1 = \text{“With insulin-dependent diabetes,} \\ \quad \text{your body makes little or no insulin.”} \\ d_1 = \text{“The condition that you have is} \\ \quad \text{insulin-dependent diabetes.”} \end{array} \right.$$

$$(2) \quad R_{U_1} = \left\{ \begin{array}{l} \textit{rhet\_rel}(\text{ELABORATION}, a_1, d_1) \\ \textit{rhet\_rel}(\text{ELABORATION}, c_1, d_1) \\ \textit{rhet\_rel}(\text{JUSTIFICATION}, c_1, d_1) \\ \textit{rhet\_rel}(\text{ELABORATION}, b_1, c_1) \end{array} \right.$$

The KB in (3) contains three semantic units among which five rhetorical relations hold (4).

$$(3) \quad U_2 = \left\{ \begin{array}{l} a_2 = \text{“We go to the bookstore.”} \\ b_2 = \text{“We go to Sam's bookstore.”} \\ c_2 = \text{“You come home early.”} \end{array} \right.$$

$$(4) \quad R_{U_2} = \left\{ \begin{array}{l} \textit{rhet\_rel}(\text{ELABORATION}, b_2, a_2) \\ \textit{rhet\_rel}(\text{CONDITION}, c_2, a_2) \\ \textit{rhet\_rel}(\text{CONDITION}, c_2, b_2) \\ \textit{rhet\_rel}(\text{MOTIVATION}, a_2, c_2) \\ \textit{rhet\_rel}(\text{MOTIVATION}, b_2, c_2) \end{array} \right.$$

To increase readability, the semantic units are given in textual form, but one should understand that a chosen formal language is actually used. The rhetorical relations are represented as first-order predicates whose first argument denotes the name of the rhetorical relation, and whose second and third arguments denote the satellite and the nucleus that pertain to that relation.

In this paper, we show how one can derive text plans from inputs of the kind shown in (1) – (2) and (3) – (4).

Key concepts

The foundations of our approach rely on what we believe to be an under-exploited part of Mann and Thompson's rhetorical structure theory (RST) (1988) and on our formalization of RST (Marcu 1996). During the development of RST, Mann and Thompson noticed that rhetorical relations exhibit strong patterns of ordering of their nuclei and satellites, which they called *canonical orderings*. The key idea of the bottom-up approach to text planning is to formalize both the strong tendency of semantic units that could be associated with the nuclei and satellites of various rhetorical relations to obey a given ordering; and the inclination of semantically and rhetorically related information to cluster into larger textual spans (Mooney *et al*. 1990; McCoy and Cheng 1991). In other words, the bottom-up approach to text planning assumes that global coherence can be achieved by satisfying the local constraints on ordering and clustering and by ensuring that the discourse tree that is eventually built is well-formed.

Uncovering the strengths of the local constraints that characterize coherent texts

The canonical orderings provided by Mann and Thompson (1988, p. 256) do not cover all rhetorical relations and do not provide clear-cut evidence about how "strong" the ordering preferences are. To solve this problem, we carried out an empirical study from which we derived statistical data concerning the ordering preferences of the nucleus and satellite of rhetorical relations and the "strength" of these preferences. We also derived data concerning the distance between the nucleus and satellite of rhetorical relations (in some cases, the nucleus and the satellite need not be adjacent). Since discourse markers are good indicators of rhetorical relations (Knott and Dale 1996) we decided to use such markers for determining the corpus of data on which we performed the study.

Materials

We used previous work on cue phrases to create an initial set of more than 450 potential discourse markers. For each potential discourse marker, we then used an automatic procedure that extracted from the Brown corpus a set of text fragments, each containing a "window" of approximately 200 words that encompassed the occurrence of the marker. Each occurrence was located approximately 150 words away from the beginning of the window. On average, we randomly selected approximately 19 text fragments per marker, having few texts for the markers that do not occur very often in the corpus, and up to 60 text fragments for markers like *and*, which occur frequently. Overall, we randomly selected more than 7900 texts. At the time of writing, 1600 of these texts were analyzed manually to derive the data presented below.

All the text fragments associated with a potential cue phrase were paired with a set of slots that described the type of usage of the potential discourse marker (*Sentential*, *Discourse*, or *Both*); the rhetorical relation that the cue phrase signalled, the boundaries, order, and rhetorical statuses (*Nucleus* and *Satellite*) of the textual units that are connected by the discourse marker; and the distance between the units. We discuss at length the corpus analysis and its results in (Marcu 1997); in this paper, we present only the results that are relevant for the examples that we use.

Procedure and results

For each potential marker, one analyst was presented with the set of randomly selected text fragments and their associated empty slots to be filled. The results were automatically exported into a relational database, from which we then computed, for each rhetorical relation, three kinds of statistical data:

1. A number between 0 and 1 that represents the strength of the preference ordering in which the nucleus goes before the satellite. For example, the strengths of the ordering preferences in table 1 show that 100% of the ELABORATIONS and 45% of the CONCESSIONS in the corpus realize the nucleus before the satellite.
2. A number between 0 and 1 that corresponds to the preference of rhetorically related units to cluster. The strength of this preference is computed on the basis of the average distance between the nucleus and satellite of a rhetorical relation. Values close to 0 reflect no preference for clustering. Values close to 0.5 reflect a preference for clustering into units that are realized as adjacent sentences. Values close to 1 reflect a preference for clustering into units that are realized as clauses of the same sentence. For example, the strength of the clustering preference that pertains to JUSTIFICATION reflects that, in most cases, JUSTIFICATION relations are realized as adjacent sentences. In contrast, CONCESSION relations are mostly realized as adjacent clauses.
3. A set of discourse markers that can be used by a generation system to indicate the nucleus and satellite of each relation.

Table 1 presents part of the statistical data that we derived for the rhetorical relations that we use in the examples given in this paper.

Rhetorical relation	Discourse markers associated with the satellite	Strength of the ordering preference (nucleus first)	Strength of the clustering preference
ELABORATION	—	1.0	0.55
	which	1.0	1.0
CONCESSION	*although*	0.45	0.985
JUSTIFICATION	—	0.0	0.525
CONDITION	*if*	0.5	0.9
MOTIVATION	*that way*	0.6	0.54

Table 1: Discourse marker, ordering, and adjacency preferences for a set of rhetorical relations.

Discussion

The results of our corpus analysis provide strong indications about ways to achieve local coherence. Using the data in table 1, one can determine, for example, that if an NLG system is to produce a text that consists of two semantic units for which a CONCESSION relation holds, then it would be appropriate to aggregate the two units into only one sentence and to realize the satellite first. The relation can be signalled by using the marker *although*, at the beginning of the satellite. In the case in which an ELABORATION relation holds between the two semantic units, it is appropriate to realize the units as two different sentences, with the nucleus being presented first. One can use no discourse marker or can use the marker *which* if the units can be aggregated as clauses of a single sentence.

From local to global coherence

Preamble

One way to formalize these local coherence preferences is as weighted constraints on ordering and adjacency. If one uses this approach, then coherent texts will be those that are characterized by valid text plans that satisfy "most" of the ordering and adjacency constraints. Before fleshing out the mathematics of "most", we believe that it is worthwhile to draw the reader's attention to the fact that a proper treatment of adjacency constraints is not straightforward because the corpus analysis provides data that pertains to a linear structure (the sequence of textual units), whereas text plans are tree-like structures. Our position is that a proper treatment of adjacency constraints is one that takes seriously the nuclearity properties that characterize valid discourse trees (Marcu 1996). When nuclearity is accounted for, two semantic units are considered *tree-adjacent* if they are arguments of a rhetorical relation that connects two subtrees and if the arguments are salient units in those trees. For example, if a certain claim is followed by two evidence units that are connected through a JOINT relation, it is appropriate to consider that both evidence units are tree-adjacent to the claim. Two semantic units are considered *linear-adjacent* if they are adjacent in the text that results from an in-order traversal of the discourse tree.

In order to provide a mathematical grounding for "most", we associate to each valid discourse tree T, a weight function $w(T)$. The weight of a tree is defined as the sum of the *proper weight*, $w_p(T)$ and the *contingent weight*, $w_c(T)$. The proper weight is given by a linear combination of the weights of the ordering constraints ($w_{ord}(\text{R}, T)$), tree-adjacency constraints ($w_{tree\_adj}(\text{R}, T)$), and linear-adjacency constraints ($w_{lin\_adj}(\text{R}, T)$) that are satisfied by each rhetorical relation R in the discourse structure T that is built (5).

(5) $$w_p(T) = \sum_{\text{R} \in T}(w_{ord}(\text{R}, T) + 0.5w_{tree\_adj}(\text{R}, T) + 0.5w_{lin\_adj}(\text{R}, T))$$

The coefficients that occur in (5) reflect our intuition that ordering and clustering are equally important for achieving coherence; nevertheless, extensive experiments could yield different coefficient values. To date, we have not carried out such experiments.

Since the input to the text-planning problem contains all possible relations between the semantic units given as input, it is likely that the final discourse tree will not use all these relations. However, despite the fact that some relations do not have a direct contribution to the tree that is built, some of their ordering and adjacency constraints may still be satisfied. We assume that discourse plans that satisfy ordering and adjacency contraints that are not explicitly used in the plans are "better" than those that do not, because the former may enable the reader to derive more inferences.

For a better understanding of this concept, assume, for example, that we are supposed to build a text plan from two units between which two rhetorical relations hold (R_1 and R_2). Assume also that we use relation R_1 to construct the text plan and assume that the ordering preference for R_1 is 0.5. If we consider only the proper weight of text plans, we have no way to choose between the two solutions of this problem, which correspond to the two different orderings of the units (both trees have the same weight). Assume, however, that relation R_2 has a strong preference for realizing one unit first. If our purpose is to enable the reader to derive as many inferences as possible, it would be then desirable to choose the text plan in which the ordering preference of R_2 is also satisfied. This position fully embraces Moore and Pollack's (1992) observation that both intentional and informational coherence should be accommodated by a theory of discourse. Our discourse model does not provide the means to explicitly represent multiple relations in the final discourse plans, but nevertheless, the contingent weight favors the plans that enable the reader to recover multiple discourse interpretations.

The contingent weight is given by a linear combination of the weights of the ordering and linear-adjacency constraints that are satisfied by each relation R that does not occur in the final text plan T (6).

(6) $$w_c(T) = \sum_{\text{R} \notin T}(0.25w_{ord}(\text{R}, T) + 0.25w_{lin\_adj}(\text{R}, T))$$

The coefficients that we use in (6) reflect the intuition that the contingent weight of a text plan is less important than the proper weight (5). In our current implementation, the contingent weight formula uses coefficients that are half of

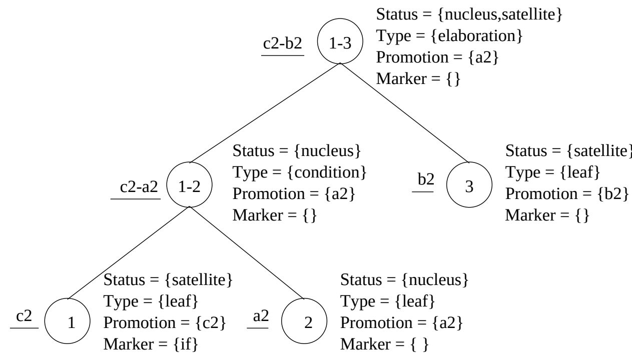

Figure 1: Example of a valid text plan for the problem in (3) – (4).

Relation	Proper		Contingent	
	Ord.	Adj.	Ord.	Adj.
rhet_rel(ELABORATION, b_2, a_2)	1.0	.275		
rhet_rel(CONDITION, c_2, a_2)	.5	.9		
rhet_rel(CONDITION, c_2, b_2)			.125	
rhet_rel(MOTIVATION, a_2, c_2)			.15	.135
rhet_rel(MOTIVATION, b_2, c_2)			.15	
Total: 3.235	2.675		.56	

Table 2: The proper and contingent weights associated with the discourse tree in figure 1.

the values of the coefficients that are used to evaluate the proper weight of a text plan.

Table 2 presents the weights that pertain to the text plan in figure 1. The text plan is represented in the notation proposed by Marcu (1996): each node in the text plan has an associated Status (nucleus or satellite), Type (rhetorical relation) and Promotion set (the set of salient units). The salient units are determined recursively; they correspond to the most important units of the immediate subspans when the relation that holds between these subspans is multinuclear, and to the most important units of the nuclear subspan when the relation is mononuclear. In addition, we associate to each node a set of markers that can be used to signal the corresponding rhetorical relation.

Given the discussion above, finding a solution to the text-planning problem corresponds to finding a discourse tree that is valid, i.e., satisfies the constraints described by Mann and Thompson (1988) and Marcu (1996), and whose weight is maximal. Since the total number of trees that can be built with a set of n units is very large ($n!4^{n-1}/\sqrt{\pi(n-1)^3})(1 + O(\frac{1}{n}))$, it is obvious that we cannot enumerate all the trees and select then those that are valid and whose weights are maximal.

Bottom-up algorithms for text planning

Algorithm 1 The simplest way to solve the text-planning problem is to generate all *valid* trees that can be built given the units in U and return those whose weights are maximal. This can be done by a variation of the Cocke-Kasami-

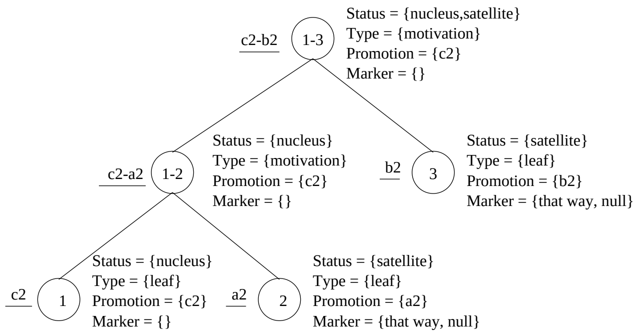

Figure 2: Example of a text plan whose weight is different from the weight of the corresponding linear plan.

Younger parsing algorithm (Younger 1967) along the lines described by Brew (1992). The algorithm starts with the initial set of n semantic units U and then constructs, at each step, all the valid trees that are made of k semantic units, where $k = 2 \ldots n$. Thus, for each k, the algorithm searches for all pairs of trees that have j and $k - j$ semantic units and builds a new tree with k semantic units if possible, i.e, if there exists a rhetorical relation $rhet\_rel(\text{R}, s, n)$ that has not been used before in any of the two trees, and if s and n belong to the salient units of the two trees. It is trivial to prove that algorithm 1 is both sound and complete, i.e., it derives only valid trees and it guarantees finding the valid tree(s) of maximal weight. Unfortunately, generating all valid solutions is still exponential in the worst case.

Algorithm 2 If one gives up on completeness, algorithm 1 can be modified such that not all valid discourse trees are generated, but only those that look more promising at every intermediate step. Algorithm 1 can be thus modified into a greedy one, which generates for every pair of trees j and $k - j$ only one tree, that of local maximal weight.

Algorithm 3 A better way to improve efficiency is by using constraint satisfaction (CS) techniques. We have constructed a CS-based algorithm that approximates first the rich tree-like structure of text plans by a linear sequence. Once the algorithm determines the sequence of semantic units that is most likely to be coherent, i.e., satisfies most of the linear ordering and adjacency constraints, it uses the algorithm described by Marcu (1996) to build a full tree-plan on the top of that sequence.

The CS-based algorithm associates initially to each semantic unit in the input an integer variable whose domain ranges from 1 to n, where n is the cardinality of U. For example, the algorithm associates to input (3) – (4) three variables, $v_{a_2}, v_{b_2}, v_{c_2}$, each ranging from 1 to 3.

For each rhetorical relation, the algorithm associates one weighted ordering and one weighted adjacency constraint along the lines described in section 3. For example, for the rhetorical relation $rhet\_rel(\text{CONDITION}, c_2, b_2)$, the ordering constraint is $v_{c_2} < v_{b_2}$ and has a weight of 0.5 and the adjacency constraint is $(v_{c_2} = v_{b_2} + 1) \vee (v_{c_2} = v_{b_2} - 1)$ and has a weight of 0.9. Since the CS-based algorithm uses only a linear representation of text plans, it is obvious that the modeling of the adjacency constraints is only an approximation of the way adjacency constraints are accounted for by algorithm 1. For example, the text plan in figure 2 has a higher weight than the weight that results from summing up all the weights of the constraints that are satisfied by the linear sequence c_2, a_2, b_2. The reason is that, in the linear sequence, the adjacency constraint that pertains to relation $rhet\_rel(\text{MOTIVATION}, b_2, c_2)$ is not satisfied because units b_2, c_2 are not adjacent in the linear sequence; however, they *are* adjacent in the resulting tree, due to the nuclearity constraints.

Since in the CS-based approach the initial target is linear, with every pair of variables the algorithm asserts also a unicity constraint; this constraint prevents two semantic units being mapped into the same value. However, if two semantic units occur as arguments of the same relations in a set R, it is impossible to distinguish between their rhetorical contributions to the text. In these cases, the unicity constraint is not asserted.

Once a constraint satisfaction problem has been derived from the input, a classical CS algorithm is employed to find out the linear sequence whose overall weight is maximal. The algorithm discussed by Marcu (1996) is then applied to this sequence and full text plans are obtained.

Implementation and experimentation

We have implemented all these algorithms in Common Lisp and tested them on a comprehensive set of examples from the NLG literature and on real problems that involve the generation of large medical texts. We have found that the first two algorithms are, on average, ill-suited for practical settings in which the number of semantic units exceeds 10. However, although finding a solution for a constraint satisfaction problem is in general NP-complete, the CS-based algorithm worked properly in practise and was easily integrated in the Sentence Planner architecture of the HealthDoc Project (DiMarco *et al.* 1995), whose goal is to produce medical brochures that are tailored to specific patients.

In fact, the semantic units in (1) are members of a large KB that encodes information that is given to diabetic patients. The semantic units that are relevant for a given patient are selected by another process. After the units have been selected, the CS-based algorithm runs and returns an ordering of the semantic units that is most likely to be coherent and the trees with maximal weight that correspond to that ordering. In this case, the discourse module proposes that in order to be coherent, the semantic units should be realized in the order d_1, a_1, c_1, b_1, which corresponds to this text:

(7) The condition that you have is insulin-dependent diabetes. Insulin-dependent diabetes is the less common type of diabetes. With insulin-dependent diabetes, your body makes little or no insulin. The pancreas, a gland found just behind the stomach, normally makes insulin.

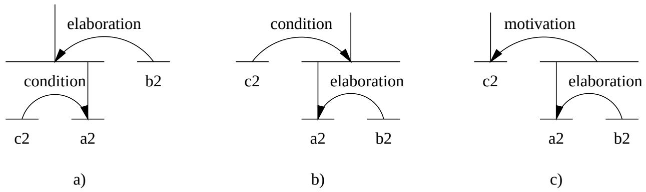

Figure 3: Valid text plans for problem (3) – (4).

Once the discourse structure for the text has been fixed, other modules operate on the semantic units in the structure. Up to this point, the following modules have been implemented: 1. Aggregation (to remove redundancies across neighboring expressions); 2. Ordering (to place clause constituents in specific positions in the emerging sentence); and 3. Reference (to plan pronominal and other reference) (see (Hovy and Wanner 1996) for a discussion). After the other modules operate on the structure that was derived by the discourse module, the resulting text is this:

(8) The condition that you have is insulin-dependent diabetes, which is the less common type of diabetes. With this condition, your body makes little or no insulin. Insulin is normally made in a gland called the pancreas found just behind the stomach.

Generating discourse plans that satisfy multiple communicative goals

By default, the algorithms introduced in this paper find plans that satisfy the goal "tell everything that is in the KB". For example, when only the default goal is used, algorithm 1 generates for the problem (3) – (4) two valid trees of maximal weight (3.375) (see figure 3.a–b).[2] However, when generating text, it is often useful to specify more than one communicative goal. In some cases, besides informing, we may also want to motivate, persuade, or deceive the reader.

Traditionally, top-down planning algorithms are given as input only one high-level communicative goal. Although we can modify the goal expansion process that characterizes top-down text planners such that the branches that use goals specified in the imput are preferred over branches that do not use such goals, we are still running into the same problem that we discussed in the beginning of the paper: there is no way to enforce that all the information that we want to communicate will be eventually included in the final text plan. In addition, the procedure described above assumes that the system can determine the communicative goal that it needs to satisfy first: after all, the system has to start the expansion process from somewhere. In the general case, such an assumption is unreasonable; and enumerating all the possible combinations is too expensive.

In contrast with top-down planning algorithms, the bottom-up text-planning algorithms can be easily adapted to generate plans that satisfy multiple communicative goals. For example, one can specify that besides conveying the information in the KB, another high-level communicative goal is to persuade or motivate the reader to come home early ($\text{MOTIVATE}(\textit{hearer}, c_2)$). Such a communicative goal can be mapped into an extra constraint that the final discourse plan has to satisfy: in this case, the extra constraint will require that the final discourse plan uses at least one rhetorical relation of MOTIVATION that takes c_2 as nucleus. When such a constraint is specified, only one tree of maximal weight (3.09) is returned by algorithm 1, that in figure 3.c. Along the lines described here, one can also specify conjunctions and disjunctions of communicative goals and pragmatic constraints that characterize ordering preferences on the linear realization of semantic units (see (Marcu 1997) for details).

As we have shown, the discourse structures that we build also contain information concerning the markers that can be used to signal various rhetorical relations. If we simply embed these markers into the final texts, we produce texts such as: "*If* you come home early, we can go to the bookstore. We can go to Sam's bookstore" for tree 3.a and "Come home early. *That way* we can go to the bookstore. We can go to Sam's bookstore" for tree 3.c. Although implementing such an algorithm is trivial, we believe that a proper account of the discourse markers should take into consideration the local lexicogrammatical constraints as well because, for example, this simple algorithm does not work adequately for the tree in figure 3.b. We are currently investigating ways in which the other modules in our system can integrate the markers suggested by the discourse module.

Conclusion

We introduced a new, data-driven approach to the text-planning problem and proposed three algorithms that can be used to map full knowledge bases into valid discourse trees. We also showed how these algorithms can be used to generate text plans that satisfy multiple high-level communicative goals and how the text plans produced by our algorithms are further refined into English text. The foundation of our algorithms relies on a corpus analysis of the local constraints that characterize the usage of rhetorical relations. The most efficient algorithm that we described here is currently used in a HealthDoc, a generation system that produces health-education materials that are tailored to particular audiences.

Acknowledgements

I am grateful to Graeme Hirst for the invaluable help he gave me during every stage of this work. I am also grateful to Chrysanne DiMarco for the stimulating atmosphere that she created in the HealthDoc group and to Eduard Hovy and Leo Wanner for their help in incorporating the ideas presented here into the HealthDoc Sentence Planner Architecture.

This research was supported by the Natural Sciences and Engineering Research Council of Canada and by Technology Ontario.

[2] Due to space limitations, we use here Mann and Thompson's (1988) representation for RS-trees.

References

Chris Brew. Letting the cat out of the bag: Generation for Shake-and-Bake MT. In *Proceedings of the International Conference on Computational Linguistics, COLING–92*, Nantes, August 1992.

J.R. Carbonell and A.M. Collins. Natural semantics in artificial intelligence. In *Proceedings of the Third International Joint Conference on Artificial Intelligence (IJCAI–73)*, pages 344–351, 1973.

Alison Cawsey. Generating interactive explanations. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, volume 1, pages 86–91, July 1991.

Chrysanne DiMarco, Graeme Hirst, Leo Wanner, and John Wilkinson. HealthDoc: Customizing patient information and health education by medical condition and personal characteristics. In *Proceedings of the Workshop on Artificial Intelligence in Patient Education*, Glasgow, Aug 1995.

Eduard H. Hovy and Leo Wanner. Managing sentence planning requirements. In *Proceedings of the ECAI–96 Workshop, Gaps and Bridges: New Directions in Planning and Natural Language Generation*, pages 53–38, Budapest, Hungary, August 1996.

Eduard H. Hovy. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63(1–2):341–386, October 1993.

Alistair Knott and Robert Dale. Choosing a set of coherence relations for text generation: a data-driven approach. In M. Zock, editor, *Trends in Natural Language Generation: an Artificial Intelligence Perspective*, pages 47–67. Springer Verlag, 1996.

Willliam C. Mann and James A. Moore. Computer generation of multiparagraph English text. *American Journal of Computational Linguistics*, 7(1):17–29, Jan-March 1981.

William C. Mann and Sandra A. Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281, 1988.

Daniel Marcu. Building up rhetorical structure trees. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI–96)*, volume 2, pages 1069–1074, Portland, Oregon, August 1996.

Daniel Marcu. *The rhetorical parsing, summarization, and generation of natural language texts*. PhD thesis, Department of Computer Science, University of Toronto, Forthcoming 1997.

Mark T. Maybury. Communicative acts for explanation generation. *International Journal of Man–Machine Studies*, 37:135–172, 1992.

Kathleen F. McCoy and Jeannette Cheng. Focus of attention: Constraining what can be said next. In Cécile L. Paris, William R. Swartout, and William C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 103–124. Kluwer Academic Publishers, 1991.

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

James R. Meehan. TALE-SPIN, an interactive program that writes stories. In *Proceedings of the Fifth International Conference on Artificial Intelligence (IJCAI–77)*, 1977.

Marie W. Meteer. *Expressibility and the Problem of Efficient Text Planning*. Pinter Publishers, 1992.

David J. Mooney, Sandra Carberry, and Kathleen F. McCoy. The generation of high-level structure for extended explanations. In *Proceedings of the International Conference on Computational Linguistics, COLING-90*, volume 2, pages 276–281, Helsinki, 1990.

Johanna D. Moore and Cécile Paris. Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics*, 19(4):651–694, 1993.

Johanna D. Moore and Martha E. Pollack. A problem for RST: The need for multi-level discourse analysis. *Computational Linguistics*, 18(4):537–544, 1992.

Johanna D. Moore and William R. Swartout. A reactive approach to explanation: Taking the user's feedback into account. In Cécile L. Paris, William R. Swartout, and William C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 3–48. Kluwer Academic Publisher, 1991.

Sergei Nirenburg, Victor Lesser, and Eric Nyberg. Controlling a language generation planner. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, volume 2, pages 1524–1530, Detroit, MI, August 1989.

Cécile L. Paris. Generation and explanation: Building an explanation facility for the explainable expert systems framework. In Cécile L. Paris, William R. Swartout, and William C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 49–82. Kluwer Academic Publishers, 1991.

Ehud Reiter. Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible? In *Proceedings of the Seventh International Workshop on Natural Language Generation*, pages 163–170, Kennebunkport, Maine, June 1994.

Roger C. Schank and Robert P. Abelson. *Scripts, plans, goals, and understanding*. Lawrence Erlbaum Associates, 1977.

R. Michael Young and Johanna D. Moore. DPOCL: A principled approach to discourse planning. In *Proceedings of the Seventh International Workshop on Natural Language Generation*, pages 13–20, Kennebunkport, Maine, June 1994.

D.H. Younger. Recognition of context-free languages in time n^3. *Information and Control*, 10:189–208, 1967.

Ingrid Zukerman and Richard McConachy. An optimizing method for structuring inferentially linked discourse. In *Proceedings of the 11th National Conference on Artificial Intelligence*, pages 202–207, Washington, DC, July 1993.