

# Domain-Dependent and Domain-Independent Rhetorical Relations

Jong-Gyun Lim

Department of Computer Science  
450 Computer Science Building  
Columbia University  
New York, N.Y. 10027  
lim@cs.columbia.edu

## 1. Rhetorical Relations as Text Planning Operators

Rhetorical relations have been used for text planning in many text generation systems ([McK82] [Hov88] [Moo89], among others), but how they are used vary rather significantly from one text planner to another. While studying them in detail side by side ([Lim92]), I have observed the following:

1. A plan operator in a typical AI planner is to carry out an action whereas that in a text planner is to inform rhetorical relations among the actions and objects.
2. A goal (or intention) in an AI planner is to change the state of the world whereas that in a text planner is to change the mental state of the hearer.

More precisely, I was led to believe in the analogy between an text planning and a typical AI planning task as shown in Table 1.

Intentions are goals of the text planner that can be realized by planning a text in terms of the rhetorical relations. Thus, the role of the rhetorical relations is to manipulate the mental objects in the mind of the hearer by creating or altering the relations among them.

With this analogy, I view rhetorical relations as realization of intentions. Furthermore, this analogy leads me to believe that rhetorical relations should be as numerous and varied as there are relations among actions and objects in the domain. This view seems to be in conflict with Dale's view (in this proceedings) which argues that rhetorical relation should only include textual relations rather than mirroring domain relations. Assuming many-to-many relation between intentional and informational

AI Planning	Text Planning
operators (actions)	rhetorical relations
state of the world	mental states
goals	intentions
plan (network of actions)	texts (network of rhetorical relations)

Table 1: Comparison Between Text Planning and A Typical AI Planning

levels (see [MP92] and Korelsky and Kittredge in this proceeding), however, I think the number of rhetorical relations should depend on the number of relations in the domain. For example, if there is a domain relation  $R$ , then depending on who his hearer is and what his intentions are, a speaker may choose different method  $M$  to inform  $R$  to the hearer. The method  $M$  may be planned in terms of several rhetorical relations or a single rhetorical relation may be used for more than one method like  $M$ . More on this is discussed later.

This analogy is in line with Traum's position of viewing rhetorical relation as speech acts or actions in general (in this proceedings). In particular, this view agrees with his point that a relation can be planned, performed, and recognized. Since actions are unbounded so should be the relations, which concurs with his criticism that it is meaningless to find the boundary for the right set of rhetorical relations.

Traum, however, tends to emphasize the importance of intentions so much as to indicate that the role of rhetorical relations is only secondary and may not even be necessary in communicating intentions. Very often (especially in casual conversation), rhetorical relations are implied and thus hidden from the surface form (e.g. (2a) and (2b) in Traum's). However, it is clear that a coherent text is structured with some meaningful rhetorical relations among its segments. Therefore, without identifying those relations (hidden or otherwise) planned by the speaker, we cannot say that the hearer understood the speaker. Thus, I take the position that in both text planning and recognition, rhetorical relations play a primary role in communicating intentions.

## 2. Need for Domain-Dependent Rhetorical Relations

Typical AI planner represents actions and objects in a hierarchical knowledge base where both domain-dependent and domain-independent concepts are represented. Hence, it is possible to plan at abstract level and generate plans in terms of the abstract plan operators. Similarly, rhetorical relations and intentions in text planning may include domain-independent types as well as domain-dependent ones.

Without the domain-dependent counterpart, however, abstract rhetorical relations by themselves are not very useful except for the limited use of meta conversation (talking about abstract relations among abstract concepts.) For example, rhetorical relations in TEXT [McK82], Hovy's RST plan operators [Hov88], and Moore's RST

plan operators [Moo89] are all domain-independent rhetorical relations. Thus, they all apply some facilities to the domain-independent relations to generate information about domain-dependent relations.

In TEXT, a domain-independent rhetorical predicate of a schema is implemented with a detailed predicate function that contains directions to search through the knowledge base to retrieve the domain-dependent relations. Unfortunately, this makes it hard to write new schemas. Hovy's RST plan operators, on the other hand, are easier to write because no domain-dependent plan operators need to be implemented. However, the burden is shifted to the text structuring process where selected pool of propositions must be interpreted and matched to some RST plan operators. This necessitates the encoding of knowledge that maps a domain specific proposition to a domain-independent rhetorical relation which in my opinion is as hard as writing domain-dependent RST relations.

Unlike Hovy's, Moore's plan operators contain intentional goals which make it possible to generate structured network of RST relations using hierarchical planning. However, an instantiated domain-independent relation typically does not capture the specific relations of the instantiated information. For example, the plan operator PERSUADE-USER-TO-DO-ACT will be instantiated to (PERSUADED ?user (GOAL ?user (DO ?user ?act))) but same strategy will be used to persuade an ?user to do an ?act regardless of who the ?user or what the ?act might be. In reality, different persuading strategy might be needed for persuading John to replace SETQ to SETF and persuading John to jump off from an airplane.

To solve this problem, Elhadad takes another extreme approach and introduces RST-like plan operators called topoi [Elh92]. Topoi are functionally equivalent to Moore's RST plan operators except that they are completely domain-dependent plan operators. The main advantages of using all domain-dependent rhetorical relations in text planning is that one does not need to struggle with naming and matching a specific relation to a generic level relation (like Evidence and Justifications) while still being able to build a text structure that reflects the rhetorical structure among the discourse units. The disadvantage is that the recursively embedded structural relations that are captured by RST relations among the various levels of discourse units can not be captured by topoi. As far as text planning is concerned, however, that doesn't seem like a disadvantage because such an information is not necessary for generating coherent text nor for responding to follow-up questions. Also, coherence of a text seems to depend more on the underlying domain plans than on the abstract rhetorical relations.

### **3. Where Are the Domain-Dependent Rhetorical Relations?**

These observations lead me to conclude that the types of rhetorical relations and intentions that are most useful for text planning are those that are needed by the text planner the most, and it seems that the most needed rhetorical relations are the domain-dependent relations that capture the specific relations between the two discourse units. How then do we collect these domain-dependent rhetorical relations? My research is to collect these domain-dependent relations from a multi-agent pro-

gramming environment [KP88] and make them available to different kind of text planners that use them to generate explanations to the agents in that environment. From this domain, I have learned two things. One is that the domain-dependent relations have to be collected from a hybrid sources including rules in expert system, constraints in a scheduler, programs, and variables in the programs. The other observation is that one planning method is not sufficient to handle the various text planning tasks. Therefore, my system allows both the application programs and the users to define domain-dependent rhetorical relations and strategies to generate explanation for those relations. Depending on the nature of planning task, one planning method might be better suited than another. Thus, my system has the flexibility to choose among the different planning strategies. So far, domain-dependent rhetorical relations have been sufficient for this text planning needs.

## References

- [Elh92] M. Elhadad. *Using Argumentation to Control Lexical Choice: A Unification-based Approach*. PhD thesis, Columbia University, New York, NY, July 1992.
- [Hov88] E. H. Hovy. Two types of planning in language generation. In *Proceedings of the 26th ACL Conference*, pages 179–186, Buffalo, June 1988. ACL.
- [KP88] P. H. Kaiser, G. E. Feiler and S. S. Popovich. Intelligent Assistance for Software Development and Maintenance. *IEEE Software*, pages 40–49, May 1988.
- [Lim92] J. G. Lim. *Planning in AI and Text Planning in Natural Language Generation*. Technical Report CUCS-038-92, Columbia University, New York, NY, July 1992. area paper.
- [McK82] K. R. McKeown. *Generating Natural Language Text in Response to Questions About Database Structure*. PhD thesis, University of Pennsylvania, New York, NY, May 1982. Also a Technical report, No MS-CIS-82-05, University of Pennsylvania, 1982.
- [Moo89] J. D. Moore. *A Reactive Approach to Explanation in Expert and Advice-Giving Systems*. PhD thesis, University of California, Los Angeles, Los Angeles, CA, 1989.
- [MP92] J. Moore and M. Pollack. A problem for RST: The Need for Multi-level Discourse Analysis. *Computational Linguistics*, 18(4):534–544, 1992.