Monitoring Conceptual and Linguistic Decisions to Generate Coherent Text

Michal Blumenstyk
Kathleen R. McKeown

CUCS-260-87

Table of Contents

Monitoring Conceptual and Linguistic Decisions to Generate Coherent Text

Michal Blumenstyk
Kathleen R. McKeown
Dept. of Computer Science
450 Computer Science
Columbia University
New York, N.Y. 10027

ABSTRACT: The extent and type of interaction that must occur between conceptual and linguistic decisions in a language generation system has been the subject of much debate. In this paper, we present general principles which can account for two classes of examples where interaction must occur. We identify how *lexical choice* and the choice between (passive with and without agent) influences text structure, determining the order of two consecutive propositions. Our account of the interaction allows for monitoring of the generation process so that an ordering of propositions can be retracted if lexical choice or syntactic construction requires a different ordering. Our approach generalizes across domains and we have implemented it as part of a domain independent language generator which is presented here.

1 Introduction

Recently there has been a flurry of controversy in language generation work regarding the extent and type of interaction that must occur between conceptual decisions (what should be said) and linguistic decisions (how to say it) (e.g. Danlos 84; McDonald 80; McKeown 85; Hovy 85; Appelt 81). At one extreme, researchers such as Danlos claim that there are *no general principles* which can be used by a language generation system because of the extensive and unpredictable interaction that must occur between conceptual and linguistic decisions. Such a claim entails that for each new domain in which generation is required, a new system must be constructed.

In this paper, we counteract such claims by presenting general principles which can account for two classes of examples where interaction between conceptual and linguistic decisions must occur. These principles capture cases where lexical choice (a linguistic decision) interacts with choices about order of information (a conceptual decision) and cases where syntactic form interacts with order. An implementation of these principles as rules in a domain independent language generator is also discussed and sample output provided.

2 The Theory

One of the factors in producing a smooth and coherent text is the manner in which the information conveyed by it is distributed. An overall plan for textual structure (e.g., McKeown 85, Mann 84, Kukich 84, Paris 85, Goguen, Linde and Weiner 82) is one way a generation system can achieve an appropriate distribution of information, but surface choice also plays a role. In this paper, we present two relations for text structure and show how lexical choice and the choice between passive with and without agent influences the choice of structure.

The relations we present hold for two-way predicates such as **act-result** and **cause-effect**. We found that either one predicate could be subordinate to the other (which we term the **main-detail** relation) or the two predicates are equally emphasized (which we term the **balanced** relation). The **main-detail** relation refers to cases where one predicate is emphasized; it establishes the **main** point and the other proposition gives additional **details**. This structural relation imposes an order on the propositions. The proposition which expresses the **main** part should precede the **detail**. The **balanced** relation, on the other hand, does not impose any order on the propositions. Either one can occur first.

Our thesis is that surface choice can interact with the ordering of the text by indicating emphasis. First, we observed that different lexical items have different information content. If two verbs are descendants of the same node, i.e, they can replace the same semantic primitive, their information contents are compared by the degree of implicature possible by using each. A verb conveying more implicature indicates emphasis. Another relationship exists between the two main verbs of the propositions in two-way predicates. These verbs usually do not descend from the same primitive node. In this case, one verb establishes the main point, while the other comes as a subordinate. Second, a syntactic construction which enables the explicit mentioning of an agent (like passive with agent) might also indicate emphasis. In any of these cases, if emphasis is indicated, the **main-detail** relation is required and the emphasized proposition must occur first.

Our account of the interaction between relations and lexical choice allows for monitoring of the language generation process in a simple way. The system initially chooses a structural relation and attempts to proceed accordingly. If it later finds that the lexical items it must use contradict its earlier choice of structural relation, that decision can be retracted and a new structural relation selected.

We would like to emphasize that the theory works on the utterance in isolation, (i.e., it is not aware of the surrounding context). That is, the theory enables the generator to make decisions in the absence of any external information. If such information is available, it should be given precedence over the decisions made according to the theory. In many cases a generator does not possess such guiding external information, and in these occasions it is very helpful to have our theory.

In the following sections, we first show in some detail how lexical choice and the choice between passive with or without agent influence the ordering of the text. We base our discussion on examples drawn from Danlos (84) to show that our principles do effectively cover her cases. We then discuss the implemented generator and conclude with some discussion of unresolved problems.

3 Lexical Choice and Order of Information

The examples that Danlos studies include sequences where choice of verbs interacts with choice of sentential order. She presents the examples shown in Example 1 as evidence for her claim that one can not decide whether or not the **act** (in this case, shooting) should precede the **result** (killing) independently of the choice of verb for the **result** (*kill*, *assassinate*, or *murder*).[1] Danlos concludes from these examples that there are no general rules a language generator can use to capture these cases and that the information must be given explicitly as input.[2]

Example 1:

1. (a) John killed Mary. He shot her.
   (b) John shot Mary. He killed her.

2. (a) John assassinated the Pope. He shot him.
   * (b) John shot the Pope. He assassinated him.

3. (a) John murdered Mary. He shot her.
   * (b) John shot Mary. He murdered her.

But note that of vocabulary falling into the same class (e.g., *kill, assassinate, murder*), some words are more specific than others. The reader can draw more inferences from *assassinate* and *murder* than from *kill*. *Kill* indicates *cause to die*, while *murder* indicates *cause to die* plus *intention*, and *assassinate* indicates *cause to die* plus *intention* and *famous target*. A speaker uses a more specific verb if s/he intends that such inferences be communicated. Hence, the use of a proposition with a more specific verb when a less specific verb could have been used implies that this is what the speaker wants to emphasize: this is the main point. Since one proposition is emphasized and the other is subordinate to it, the **main-detail** relation is in effect. If we initially chose the **balanced** relation ordering the **act** first, we must retract that earlier decision and reverse the order of the text.

Examples from other domains support this generalization. For **cause-effect** sequences that we considered, either order of propositions is acceptable if a less specific verb is used for the effect. A more specific verb indicates emphasis, however since more inferences can be drawn. Again, the **main-detail** relation must be used since there is no longer a balance between propositions. Examples in this domain are shown below in Example 2.

Example 2:

1. (drove/raced)
   (a) Yesterday the trains weren't running. I drove to work.
   (b) I drove to work yesterday. The trains weren't running.
   (c) I raced to work yesterday. I left a little late.
   *(d) I left a little late yesterday. I raced to work.

2. (walked/slipped/sauntered)

[1] Note that Danlos does not present Example 1.1 exactly as shown here for the case of the verb *to kill*, but from her other examples involving *to kill* and *to shoot*, we assume that she would agree that these sequences are acceptable.

[2] Danlos uses a discourse grammar that specifies all possible legal combinations of verbs, order, and syntactic form.

(a) I walked to the store. I needed a quart of milk.
(b) I needed a quart of milk. I walked to the store.
(c) I slipped past the store. I didn't want the cashier to see me.
*(d) I didn't want the cashier to see me. I slipped past the store.
(e) I sauntered past the store. I wanted to appear casual.
*(f) I wanted to appear casual. I sauntered past the store.

Finally, we note that context and/or speaker intention may overrule the choice of a semantically appropriate verb or the re-ordering of sentences. If the speaker intends to emphasize killings, the verb *kill* may be selected over *assassinate* even if the victim is famous as in (1), Example 3. If context indicates the implications of a specific verb are already known, then the use of that verb does not imply emphasis and either order is acceptable as in (2), Example 3. Or, if semantics requires the use of *assassinate* and the speaker absolutely intends that the **result** come last, the unusual order can be linguistically marked (for example with stress as in (3), Example 3). Our approach provides the hooks for incorporating such an influence by allowing for the retraction of earlier decisions.

Example 3:

1. There have been a lot of killings at the vatican lately. First the guard was accidentally pushed off the balcony and killed. Then the bishop was killed when a chandelier fell on his head. After that, a thief killed the monsignor during a burglary. Finally, John killed the Pope last Thursday.

2. Why were you in such a rush?
I left a little late so I raced to work.

3. John shot the Pope. He *assassinated* him.

4 Syntactic Construction and Order of the Propositions

In this section we show how the theory handles interaction between the syntactic construction agent or agentless passive and order of the propositions.

First, let us examine one of the examples from Danlos(84) which deals with such an interaction. The semantic representation Danlos' example used is as follows:

```
(SEM1)  ACT :   Shooting
                agent : John
                goal : Mary
        RESULT:   state : Dead
                object : Mary
```

Ignoring lexical choice and other linguistic and conceptual decisions, and concentrating on decisions involving the choice of passive and the order of propositions[3], Danlos listed 4 possibilities for combining the ACT and the RESULT:

[3]To simplify the discussion, we made the assumption that every proposition is realized as a sentence, thus we use the terms *proposition* and *sentence* interchangibly.

Example 4:

(a) Mary was killed by John. She was shot.
(b) Mary was killed. She was shot by John.
(c) Mary was shot by John. She was killed.
* (d) Mary was shot. She was killed by John.

As one might observe, the 4 possibilities are the combinatorial result of having 2 variables (order of text units and syntactic construction) each capable of assuming 2 values (**act** followed by **result** or vice versa for the order, agentless passive vs. passive with agent for the syntactic construction). According to Danlos, the awkwardness of (d) and the "goodness" of (a) - (c) are not calculable from the data. Therefore, the generator must be provided with the explicit information of which combination is acceptable and which is not.

Now, let us analyze Example 4 in light of our theory. In the previous section we dealt with relationships between verbs when they replace the same semantic primitive. Here we are concerned with a relationship between the two verbs which function as the main actions of a two-way predicate. Usually one verb indicates the main point which the speaker would like to convey. This verb implies a set of possible actions and the second verb is a member of this set. That is, the second verb fills a slot which the first one creates. We term the verb which indicates the main point as the *indicator* and the second verb which fills the created slot as the *filler*. In Example 4 the two main verbs are *kill* and *shoot*. *Kill* is the indicator while *shoot* is the filler, since *kill* implies that there was some action which caused the killing, and *shoot* describes such an action. The relationship between the two main verbs is not the only factor which influences the relation between two propositions. The presence or the absence of an explicit agent is another such factor. Keeping everything else equal, a sentence with an explicit agent contributes more to the establishment of the main point than a sentence with an implicit agent.

Bearing the above facts in mind, let us examine (a)-(d) closely. In the first sentence of (a), "Mary was killed by John", the main verb is an indicator while the verb "to shoot" is the filler. In addition, the first sentence explicitly mentions the agent, which is absent from the second sentence. Thus the first sentence establishes the main point, while the second is subordinate as it provides additional details. Such a phenomenon fits the **main-detail** structural relation. Recall that this relation imposes an order on the propositions. The proposition which establishes the main point should come first, while the sub-ordinate proposition comes second. The order requirement is satisfied by combination (a) and therefore it sounds natural. Observe that (d) uses the same sentences as (a), but in the reverse order. Thus (d) first gives sub-ordinate details and then establishes the main point. According to our theory, the **main-detail** relation is violated and thus the produced text sounds odd. Therefore, if one follows our theory, the goodness of (a) and the awkwardness of (d) are predictable and calculable from the data.

(B) and (c) of Example 4 can be analyzed in a similar fashion. "Mary was killed" has an indicator as a main verb where "She was shot by John" has a filler verb, but the latter has an explicit agent while the former does not. Consequently, each sentence is equally emphasized and neither dominates the other. The **balanced** information relation characterizes these examples. This relation does not impose any order on the propositions, so we can have any sentence in the first position. Again, our theory accounts for the observed phenomenon explaining why both (b) and (c) sound coherent.

The following examples show that our analysis holds in other domains as well. They fit the same pattern as Example 4 and thus are handled identically by the theory, regardless of the semantic domain from which they are drawn.

Example 5:

(a') Mary's condition was improved by the doctor. She was injected with medication.
(b') Mary's condition was improved. She was injected with medication by the doctor.
(c') Mary was injected with medication by the doctor. Her condition was improved.
* (d') Mary was injected with medication. Her condition was improved by the doctor.

Observe that the relationship between *improve* and *inject* (the two main verbs) is the same as the relationship between *kill* and *shoot* in Example 4. That is, *improve* implies that there is a set of possible actions, while *inject* is a member of that set. Using our terminology, *improve* is the indicator and *inject* is the filler. Taking into consideration the presence or absence of an explicit agent and using the same reasoning as in Example 4, we observe that (a') complies with the **main-detail** information relation, while (b') and (c') agree with the **balanced** relation, thus (a') through (c') all sound coherent. (d') violates the **main-detail** relation, and as our theory predicts, such a violation produces an awkward text.

Example 6 below is similarly explained by the theory.

Example 6 :

(a") The stew was burned by Danny. It was left on the burner.
(b") The stew was burned. It was left on the burner by Danny.
(c") The stew was left on the burner by Danny. It was burned.
* (d") The stew was left on the burner. It was burned by Danny.

5 GEN - the Planner/Generator system

In this section, a computer program, GEN, which successfully implements the theory is described. GEN is written in PLNLP[4] (Heidorn 72), a computer language which can be used for both parsing and generation. PLNLP enables the user to write augmented phrase structure rules which manipulate the basic data structure of the language - the PLNLP record. We wrote GEN using PLNLP Encoding rules, which are used by the *Encoding*

[4] Programming Language for Natural Language Processing

Algorithm and manipulate the records in a top-down sequential fashion. In our system, the program starts with one record (the semantic input) and transforms this record by application of the rules until a complete surface string is produced.

GEN is composed of four modules:

1. The Augmented Lexicon.
2. The Planner.
3. The Lexical Choice module.
4. The Surface Generator.

The Augmented Lexicon is coded in Named PLNLP records. It contains knowledge organized in a network hierarchy, where the leaves of the hierarchy are the lexemes. The lexicon carries syntactic and semantic data for each lexeme, and also the Information Content of the lexeme, which is used for the implementation of the theory.

The knowledge embodied in modules 2-4 is expressed in PLNLP encoding rules. The Planner is responsible for producing the plan of the utterance. It performs its task by examining the semantic input and consulting the augmented lexicon and the structural relations. The Lexical Choice module examines the semantic primitive passed to it and chooses the lexical item to express this primitive. Lexical choice is a one-to-many mapping, and this module must possess enough intelligence to make the the best choice given the input parameters. The Surface Generator works on the output of the planner, which is a rough outline of the utterance. The surface generator refines the plan, performs some noun phrase planning including pronoun substitution, and uses syntax rules to produce grammatically correct surface strings. The surface generator performs most of the labor of the generation, since it carries the outlined plan to completion.

Now, let us show how a utterance is produced, concentrating on the parts of the algorithm which utilize the theory. Recall that the input to the system is the semantic representation of the intended utterance (similar to SEM1 in section 4) expressed in a PLNLP record. The planner is the first module which manipulates the input record. First, it checks to see whether the input record has an information relation, and if not it chooses one by examining discourse information such as focus or chooses a default information relation if no discourse information is available. Secondly, the planner makes some preliminary conceptual decisions, such as order of the predicates and what should be mentioned explicitly and what implicitly. Next, the planner calls the Lexical Choice module which selects lexical items for the main predicates of the record. When control returns to the planner, it has already an outlined plan which is ready to be transferred to the surface generator. But, and this is crucial, before the surface generator is called to embark on the lengthy process of producing the utterance, the planner tests whether the decisions made so far comply with the information relation. If the information rules are complied with, the surface generator is called and the generation process continues to completion. However, if a violation is detected, the planner has to revise its original plan. It does that by retracting one or more of the decisions it previously made (lexical choice or order of predicates). When a decision is first made, its basis is recorded. That is, the planner remembers why it believes in each decision. Thus, when conflicting decisions are detected, the planner retracts those decisions which have the weaker supportive evidence[5]. The revised plan is tested again for compliance with information rules, and if no violation occurs the surface generator is called and the generation process proceeds to completion.

The advantage of using the theory in the planning process is quite clear. The planner is able to detect very early in the generation process if it proceeds in the wrong direction. If this is the case, the plan is revised before the time consuming process of surface production is initiated, and more importantly, before awkward output is produced by the system.

Sample input and output of our system are shown below. Input is in the form of a PLNLP record.

Example 7:

```
Input : R500 (Act    = <'shoot', goal='John', agent='Mary'>
              Result = <state='dead', goal='John', agent='Mary'>)
```

After passing through the planner module, the record status is shown below. Note that the default value is used for the information relation.

```
PLANSEG (Inforelation = <'Balanced',default>
         Order  = <Act,Result>
         Act    = <'shoot', goal='John', agent='Mary'>
         Result = <state='dead', goal='John', agent='Mary'>)
```

After the lexical choice module has assigned the appropriate lexemes, the record status is:

```
PLANSEG (Inforelation = <'Balanced',default>
         Order  = <Act,Result>
         Act    = <'shoot', goal='John', agent='Mary',
                  Lex='SHOOT'>
         Result = <state='dead',goal='John',agent='Mary',
                  Lex='KILL'>)
```

Now the rules which are responsible for violation detection are applied to the record. The decisions which have been made so far (lexical choice, order, and information relation) are examined. Since no contradiction is found, no retraction of decisions is necessary. The balanced relation may be followed and the surface generator assumes control. The final output is:

Mary shot John. She killed him.

Example 8:

```
Input : R501 (Act    = <'shoot', goal='John Kennedy',
                       agent='Mary'>
              Result = <state='dead, goal='John Kennedy',
                       agent='Mary'>)
```

Initially, planning proceeds as in Example 7. The record status after preliminary planning steps is:

---

[5] A default decision is always the weakest, but there are other heuristics which enables the planner to order decision strength.

PLANSEG (Inforelation = <'Balanced',default>
Order = <Act,Result>
Act = <'shoot', **goal='John Kennedy'**,
agent='Mary'>
Result = <state='dead, **goal='John Kennedy'**,
agent='Mary'>)

The lexical choice module now processes the above record. Since "John Kennedy" has the attribute *Famous* (not shown here), the lexical choice module chooses the lexeme *assassinate* to express the change of state in the **result**. The status of the record is now:

PLANSEG (Inforelation = <'Balanced',default>
Order = <Act,Result>
Act = <'shoot', **goal='John Kennedy'**,
agent='Mary',Lex='SHOOT'>
Result = <state='dead, **goal='John Kennedy'**,
agent='Mary',Lex='ASSASSINATE'>)

This time, the planner detects a contradiction - the second proposition (the **result**) contributes more to the main point because of the use of "assassinate", which means that the **balanced** relation has been violated. The planner examines the decisions, and since it is the least confident in the value it assigned for the information relation (the default was used), this decision will be retracted. The planner utilizes the information it has acquired during the exploration of the erroneous path to assign a different value for the information plan. A change in the information relation calls for a revision in the order decision, and now the record status is as follows:

PLANSEG (Inforelation = 'Main-Detail'
Order = <Result,Act>
Act = <'shoot', **goal='John Kennedy'**,
agent='Mary',Lex='SHOOT'>
Result = <state='dead, **goal='John Kennedy'**,
agent='Mary',Lex='ASSASSINATE'>)

The revised plan was checked for violation, and since none has been detected, the generation process proceeds normally to yield the following output. Note that the propositions are reordered.

Output : Mary assassinated John Kennedy. She shot him.

The next 2 examples are from a different domain and are handled by GEN in the same manner as examples 7 and 8, demonstrating the generality of the rules.

Example 9:
Input : R503 (Act = <'go', agent='Donna',
manner = <direction='down',
location='stairs'>>
Result = <'exit', agent='Donna', goal='house'>)

Output : Donna went down the stairs. She left the house.

Example 10:
Input : R504 (Act = (the same as in Example 9)

Result = <'exit', agent='Donna', goal='house', emotion='fear'>)

The additional fact (emotion='fear') caused the lexical choice module to chose ESCAPE instead of LEAVE, and the same chain of events as in example 8 was triggered. Again, note that the propositions are reordered. The resulting output is as follows:

Output : Donna escaped the house. She went down the stairs.

6 Related Research

While a number of researchers (e.g., McDonald 80; McKeown 85; Derr and McKeown 84; Mann 84) have assumed separate components for determining what to say and how to say it in order to focus on problems in one component or the other, other researchers (e.g., Appelt 81; Danlos 84; Hovy 85) have developed systems that crucially rely on a single integrated component. Our work straddles these two approaches by allowing for modularization of the two components (and thus, improving portability of the system between domains) but by specifying particular points at which interaction between the two components must occur. At these points, decisions made by either the conceptual component or linguistic component can be retracted and new decisions made. Because we specify the kind of interaction that must occur, our work is probably most similar to Ritchies's (84) who calls for interaction between the two components on constructing noun phrases, but we focus instead on interaction for lexical choice of verbs, passive with and without agent, and order of propositions.

Because we make use of the notion of structural relation, a brief comparison with work on text structure, rhetorical relations, and coherence relations (e.g., McKeown 85; Mann 84; Paris 85; Kukich 84; Hobbs 78) is called for. In contrast with these other works, we do not present plans for textual sequences, nor do we make any claims about the coverage of our relations. There may, in fact, be many more such relations. We simply note that the **main-detail** and **balanced** relations do exist in natural text sequences. Our contribution is the identification of how surface constructions influence the creation of these relations between two consecutive sentences.

Finally, we note that our relations are similar to the two structural relations, **dominance** and **satisfaction-precedes**, described by Grosz and Sidner (85). In fact, the **main-detail** relation appears to be equivalent to the **dominance** relation. While the **balanced** relation is like the **satisfaction-precedes** relation in that it can be thought of as specifying the relation between two siblings in a larger hierarchical structure, the **balanced** relation does not imply any ordering between the two siblings, while **satisfaction-precedes** does.

7 Open Research Problems

There are still many open problems which need to be explored. In this paper we discussed only two relations for text structure. There may be more relations which are influenced by surface choice in different ways. Another open problem is how our theory propagates to larger text units. Can the same relationships which hold in two-way predicates also hold between paragraphs? Alternatively, the relations we present may be part of larger text plans and the re-ordering dictated by surface choice may allow for internal re-ordering of otherwise fixed text plans.

A different problem concerns the priority of external contextual information. Currently, we point out that context may override any decision made by the theory, but there might be cases as well where the theory should be given priority. Lastly, there is the problem of interaction between different surface constructions. We have dealt here just with a subset of the possible linguistic and conceptual decisions. Further study should be applied to explore the interactions between a larger set of such decisions.

8 Conclusions

We have shown here how a theory relating surface choice and structural relations can be used to monitor the process of language generation to ensure that the produced text is natural and coherent. We have specified how two types of surface choice influence the relation between consecutive sentences in a general way. This means that conceptual and linguistic choices during planning/generation can be decided independently if the system has facilities to examine the decisions, to detect contradicting ones, and then to retract the erroneous choices. Since our theory generalizes across domains, we have been able to implement it as part of a domain independent generation system. Thus our work allows the planning/generation process to be performed in a general fashion.

References

(Appelt81). Appelt, D.E., Planning Natural Language Utterances to Satisfy Multiple Goals, Ph.D. dissertation. Stanford University, Stanford, Ca., 1981.

(Danlos84). Danlos, L., "Conceptual and Linguistic Decisions in Generation", in *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford University, Ca. July 1984. pp. 501-504.

(Derr and McKeown 84). Derr, M.A. and McKeown, K.R., "Using Focus to Generate Complex and Simple Sentences", in *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford University, Ca. July 1984. pp.319-326.

(Goguen, Linde and Weiner 82). Goguen, J.A., Linde, C. and J.L.Weiner, Reasoning and natural explanation, unpublished proceedings of ISI Workshop on Explanation, 1982.

(Grosz and Sidner 85). Grosz, B.J. and C. L. Sidner, "Discourse Structure and the Proper Treatment of Interruptions," in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, Ca., August 1985. pp. 832-9.

(Heidorn 72). Heidorn, G.E., "Natural Language Inputs to a Simulation Programming system", Technical Report NPS-55HD72101A, Naval Postgraduate School, Monterey, Ca., 1972.

(Hobbs 78). Hobbs,J., Coherence and Coreference. SRI Technical Note 168, SRI International, Menlo Park, Ca., 1978.

(Hovy 85). Hovy, E.H., "Integrating Text Planning and Production in Generation," in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, Ca., August 1985. pp. 848-51.

(Kukich 84). Presentation at the 2nd Annual Workshop on Language Generation, Stanford, Ca., 1984.

(Mann 84). Mann, W.C., "Discourse Structures for Text Generation", in *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford University, Ca. July 1984. pp. 367-375.

(McDonald 80). McDonald, D.D., Natural language production as a process of decision making under constraint. Ph.D dissertation, MIT, Cambridge, Mass. (1980).

(McKeown 85). McKeown, K.R., *Text Generation: Using discourse strategies and focus constraints to generate natural language text*, Cambridge Univ. Press, Cambridge, 1985.

(Paris 85). Paris, C.L., "Description Strategies for Naive and Expert Users," in *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, Chicago, Ill., July 1985. pp. 238-45.

(Ritchie 84). Ritchie, G., "A Rational Reconstruction of the Proteus Sentence Planner", in *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford University, Ca. July 1984. pp. 327-329.