Investigating the structure of procedural texts: identification of titles and instructions

Estelle Delpech, Patrick Saint-Dizier

IRIT-CNRS – 118, route de Narbonne 31062 TOULOUSE – France

Abstract

This paper presents ongoing work dedicated to parsing the textual structure of procedural texts. We propose here a model for the intructional structure and criteria to identify its main components: titles, instructions, warnings and prerequisites. The main aim of this project, besides a contribution to text processing, is to be able to answer procedural questions (How-to? questions), where the answer is a well-formed portion of a text, not a small set of words as for factoid questions.

Résumé

Nous présentons ici un travail en cours dédié à l'analyse syntaxique et sémantique de la structure textuelle des textes procéduraux Nous proposons ici un modèle et des critères pour identifier les principaux composants de cette structure : titres, instructions, avertissements et pré-requis. L'objectif principal du projet, à côté d'une contribution à l'analyse textuelle est de permettre de répondre à des questions procédurales dans le cadre des systèmes question-réponses en Comment faire ? où la réponse est une portion bien formée de texte et non pas une information ponctuelle comme c'est le cas pour les questions factoïdes.

Mots-clés : syntaxe et sémantique textuelle, systèmes question-réponses.

1. Situation and Aims

The main goal of this work is to be able to answer procedural questions, which are questions whose induced response is typically a fragment, more or less large, of a procedure, i.e., a set of coherent instructions designed to reach a goal. Recent informal observations from queries to Web search engines show that procedural questions is the second largest set of queries after factoid questions (de Rijke, 2005).

Answering procedural questions thus requires to be able to extract not simply a word in a text fragment, as for factoid questions, but a well-formed text structure which may be quite large. Analysing a procedural text requires a dedicated discourse analysis, e.g. by means of a grammar. Such grammars are not very common yet due to the complex intertwinning of lexical, syntactic, semantic and pragmatic factors they require to get a correct analysis. Discourse grammars have basically a top-down organization, they take discourse acts as their basic units, instead of just words, they account for the structure and for the interactions between these acts and they require a relatively elaborated conceptual representation as output. Such a grammar must capture the discourse cohesion, possibly the communicative intentions, as well as the discourse organization, e.g. in terms of plans.

Procedural texts are organized sets of instructions, they may also be sets of advices, as in social behavior texts. In our perspective, procedural texts range from apparently simple cooking recipes to large maintenance manuals. They also include documents as diverse as

teaching texts, medical notices, social behavior recommendations, directions for use, assembly notices, do-it-yourself notices, itinerary guides, advice texts, savoir-faire guides etc. Even if procedural texts adhere more or less to a number of structural criteria, which may depend on the author's writing abilities and on traditions associated with a given domain, we observed a very large variety of realisations, which makes identifying the structure of such texts quite challenging.

Procedural texts explain how to realize a certain goal by means of actions which may be temporally organized. Procedural texts can indeed be a simple, ordered list of instructions to reach a goal, but they can also be less linear, outlining different ways to realize something, with arguments, advices, conditions, hypothesis, preferences. They also often contain a number of recommendations, warnings, and comments of various sorts. The organization of a procedural text is in general made visible by means of linguistic and typographic marks. Another feature is that procedural texts tend to minimize the distance between language and action. Plans to realize a goal are made as immediate and explicit as necessary, the objective being to reduce the inferences that the user will have to make before acting. Texts are thus oriented towards action, they therefore combine instructions with icons, images, graphics, summaries, preventions, advices, etc.

Research on procedural texts was initiated by works in psychology, cognitive ergonomics, and didactics. Several facets, such as temporal and argumentative structures have then been subject to general purpose investigations in linguistics, but they need to be customized to this type of text. There is however very little work done in Computational Linguistics circles. The present work is based on a preliminary experiment we carried out (Delpech et ali. 07), (Aouladomar 2005) where a preliminary structure was proposed. From a methodological point of view, our approach is based on (1) a conceptual and linguistic analysis of the notion of procedure and (2) a mainly manual corpus-based analysis, whose aim is to validate and enrich the former.

In this short paper, we summarize our results, focussing (1) on the conceptual notion of intructional compounds, which does capture the complexity just advocated, and (2) on the recognition of titles, instructions and instructional compounds. An quite comprehensive evaluation was carried out that we breifly report here. This work is part of the ANR TextCoop project.

2. The structure of procedural texts: Instructional Compounds

Procedural texts contain two basic structures: titles, analyzed as goals (with which questions will match), and instructions serving these goals. However, in most types of texts, we do not have just sequences of simple instructions but much more complex compounds. We noted that these compounds are organized around a few main instructions, to which a number of subordinate instructions, warnings, arguments, and explanations of various sorts are adjoined. Procedural texts also contain general purpose prerequisites and warnings, besides those included into instructional compounds.

Let us essentially, in this contribution, focus on the instructional compound structure, which is, by far, the most complex element. It has a relatively well organized discourse structure, composed of several layers, which are:

- The ***justification and explanation structure***, which has wider scope over the remainder of the compound, indicates motivations for doing actions that follow in the compound (e.g. *in your bedroom, you must clean regularly the curtains...*, which here

motivates actions to undertake). It may also indicate the goal of the instructions that follow. It may play a role quite close to the causal structure described below.

- The ***instruction kernel structure***, which contains the main instructions. These can be organized temporally or just be sets of actions. Actions are identified most frequently via the presence of action verbs (in relation to the domain) in the imperative form, or in the infinitive form introduced by a modal. We observed also a number of forms of subordinated instructions adjoined to the main instructions. These are in general organized within the compound by means of rhetorical relations, that we introduce below.
- The ***deontic and illocutionary force structures***: consist of marks that operate over instructions, outling two different parameters: (1) deontic: obligatory, optional, forbidden or impossible, alternates (or), (2) illocutionary and related aspects: stresses on actions: necessary, advised, recommended, to be avoided, etc.
- A ***causal structure*** that indicates, within the compound the use or the motivation of an instruction. We identify several types of causes, such as: intend-to (*push the button to start the engine*), instrumented (*use a 12 inch ket to dismount the equipment*), facilitation and continue (*keep the liquid warm till its color changes*).
- The ***conditional structure***: introduces conditions over instructions within the compound or even over the whole instructional compound. Some conditions may have the form of case structures, based on exclusive conditions (as in programming languages). Some conditions also introduce possible actions (if you are an expert, ...)
- The ***rhetorical structure*** whose goal is to enrich the kernel structure by means of a number of subordinated aspects (realized as propositions, possibly instructions) among which, most notably: *enablement, motivation, argument for, circumstance, elaboration, instrument, precaution, manner*. The rhetorical structure is in general composed of instructions (satellites) related to the instructions in the kernel. A specific structure, of much importance is the argumentative structure, where arguments are given to motivate the user (prevention, threats, rewards, etc. (Aouladomar and Saint-Dizier, 2005)

Let us now give an illustrative example, extracted from the 'Do-It-Yourself Home' domain:

In a bedroom, it is necessary to clean curtains. These are cleaned first with a vacuum-cleaner to remove dust, then, if they are in cotton, they can be washed in the washing machine at 60 degrees; if they are white, it is even recommended to add some bleech so that they look whiter. With some starch, they can be easily ironed.

In this text, the sequence: *In the bedroom, it is necessary to clean curtains* is analyzed as a justification of the actions to undertake. The next portion: *These are cleaned first with a vacuum-cleaner to remove dust, then, if they are in cotton, they can be washed in the washing machine at 60 degrees.* is the instruction kernel, where the last instruction is associated with a condition. Finally, *If they are white, it is even recommended to add some bleech so that they look whiter. With some starch, they can be easily ironed.* are two subordinated clauses, analyzed as advices.

Here is another example, in French, using the bracketed notation to indicate relative scope of elements:

[[Cond Si vous souhaitez préserver quelques blancs sur le papier,]

[instruction aspirez la couleur avec un chiffon sec en tissus éponge.

[facilitation [instruction Vous devez déborder un peu de la zone à préserver:

[explication la couleur peut passer sous le chiffon par capillarité]]]]].

In this example, a condition has wider scope over the whole instructional compound. The kernel is composed of a single instruction, modified by a subordinate instruction (a facilitation in rhetorical terms), itself modified by an explanation (which is not an instruction).

3. Recognizing Titles, Instructions and Instructional Compounds

Let us now develop in more depth the different phases of our system. Prelimany steps include cleaning and labelling text objects. Then title and instructions can be recognized.

3.1. Cleaning Web texts and tagging

The input of our system are raw Web pages. From these pages, we need (1) to extract relevant text, that is, any kind of text that is not navigation help, advertisements or comments posted by cybernauts (2) to select and simplify the html tags so as to keep the main typo-dispositional information (paragraph breaks, subdivisions of paragraphs into lines, lists and their subdivision into elements, emphasis). Although (2) was quite an easy task, we had some difficulties achieving (1). We designed an algorithm that returns, foreach paragraph, if its text can be considered relevant or not. It mainly uses paragraph length and proportion of close-class words criteria. We evaluated it on 100 Web pages, coming from 12 different web sites. The results were: 0,95 precision and 0,76 recall. This cleaning process is not maintained in our implementation. We evaluate the recognition of titles and instruction on a hand-cleaned corpus. Although the results seem good at first sight, one has to keep in mind that it is a preliminary step. Yielding a 0,76 recall means that only three fourths of relevant text is kept, which we consider too low for a pre-processing step. This part of the project will be udertaken by our industrial parners.

The next stage is to tag the different lexical objects of the text, so that the segmentation of titles and instructions can be done properly. For that purpose, we use the Treetagger, that labels all the objects (syntactic category, morphological features). We also add some semantic considerations about action verbs. Of particular interest to us is the recognition of verbs, some nouns and adjectives, modals and connectors of various kinds.

3.2. Recognizing Titles

For answering How-to questions it is obviously of much importance to recognize titles and possibly hierarchies of titles in complex texts. A first observation is that html encodings are, by far, not homogeneous. Titles are coded with the tag <hi> in only 20% of the cases over the 600 titles observed. In most cases the tag <b> is used, possibly also <emp>, <u> and a few others (macros...). Encodings may be quite homogeneous within a given web site, but heterogeneity prevails over different sites, even in the same domain.

We identify titles in two steps. First, the algorithm processes the paragraphs of the text one by one, and give them one of these tags: title, text or ambiguous. This first step processes easy cases. For example, an easy case for a title is a paragraph composed of a unique sequence of words, less than 12 words long and bearing emphasis. The tag text will be given without

doubt if the paragaph is subdivided into smaller units or is longer than 12 words. Ambiguous parapraphs are mainly short sequences of words (12 words or less) with no emphasis.

The second step desambiguates the ambiguous paragraphs one by one, using the tags given by the first step to its surrounding paragraphs. For example, an ambiguous paragraph between two paragraphs tagged as text will be considered a title. Similarly, an ambiguous paragraph followed by a title is labelled text. We have elaborated about 7 such rules that raises ambiguities. This second step also operates some repairs on the tags yielded by the first step. For example, any sequence marked *title* at the end of the text will be repaired as *text*. Each desambiguation/repair rule is applied sequentially and in a specific order to the list of tags.

The title hierarchy is very difficult to identify without content analysis. However, standard procedural texts are not very long and tend to be relatively linear. This means that, besides the page title, we observed in 80% of our texts not more than 2 levels of titles (exluding the main title). We observed two regular types of titles that can be correlated to some form of hierarchy. Type 1 is a title separated from its following paragraph by a <p> tag. Type 2 is a title separated from its following paragraph by a
 tag. Although we still have no means to tell the exact level titles, we can quite confidently say that a type 2 title will be at a lower level than a type 1 title, whatever the website or the domain. This information may be useful for question-title matching: type 2 titles are expected to introduce paragraphs that deal with more specific aspects of the procedure than paragraphs introduced by a type 1 title. Type 2 titles could help answering specific questions. One remaining difficulty for question-title matching is that titles have often a very elliptic structure.

3.3. Recognizing instructions and instructional compounds

While working on corpora, we noted that what is usally called an instruction ranges from clearly injonctive clauses to implicit prescriptions (this complexity is reflected in the complexity of manual annotation tasks, see below). Instructions are recognized on the basis of two factors: contents, around action verbs in certain forms to identify an instruction and typographic factors for its delimitation (beginning and end) via html tags, punctuation marks or connectors. Currently, we use a set of only 14 lexico-morphological patterns, that encompass the most prototypical ways of expressing instructions. We use lexical ressources such as action verbs, incentive verbs, nouns and adjectives. They must have in French specific forms: imperative, infinitive, modal + infinitive, dummy pronoun ‘on’ + finite verb (this pattern has a semantic restriction: only action verbs are allowed), middle reflexive constructions, and gerundive forms. The frequency usage of each of these forms largely varies across domains (e.g. coocking recipes mainly use imperative while video game solutions make high usage of the dummy pronouns ‘on’ or even of finite forms in the first person singular). The recognizer (also called the segmenter) includes 14 morphosyntactic generic patterns. The segmenter is implemented in standard Perl. Note that English seems to have a simpler set of forms while Spanish has a lot of finite forms, making instructions slightly more difficult to recognize.

Instructional compounds are composed of instructions. They are delimited as follows: by means of typographic marks: ending of enumeration (e.g. sequences) or by ‘strong’ marks in long paragraphs. These marks are in general temporal (Two hours later,...), conditional expressions or goal expressions.

Finally, a grammar, based on a simple transposition of a few Minimalist Theory principles allows us to bind all the parts of the text. The grammar runs in Prolog in our prototype. The

output is an XML file that reflects the text structure. Here is an extract of what we get before the grammar application, where terminal elements are tagged:

  <titre>Gâteau au chocolat gourmand</titre>

<prerequis>

<titre> Ingrédients</titre>

 150 g de chocolat noir

 75 g de beurre doux

 210 g de lait condensé …….</prerequis>

Utilisé depuis la nuit des temps, le chocolat .... si vous souhaitez l'inclure dans la composition de votre oeuvre .

temps: 2 heures, assez facile.

<titre> la préparation :</titre>

 <compinstr> <instr> 1. Tapisser un petit moule rectangulaire de papier aluminium.</instr></compinstr>

<compinstr> <instr> 2. A l'aide d'un couteau tranchant, concasser les amandes. </instr></compinstr>

.... <compinstr> <instr> Dans une casserole à fond épais, placer le chocolat cassé en morceaux, le beurre, le lait et la cannelle.</instr>

<instr> Chauffer doucement à feu doux pendant 3 à 4 minutes en remuant avec 1 cuillère en bois.</instr>

<instr> Bien battre le mélange. <instr> Incorporer les amandes, les biscuits et les abricots en remuant bien. </instr> .... </compinstr>

<compinstr> <instr> Au bout d'une heure, ... </instr> ..... </compinstr> .....

4. Evaluation

The evaluation we have carried out allows us to have an estimate of the overall quality and accuracy of the recognition mechanisms, outlining problems and gaps for future evolutions. From that point of view, it is an *indicative* evaluation.

4.1. Evaluation process and results

The first step was a manual annotation carried out by two independent annotators of 78 Web pages over 5 domains: coocking recipes, do it yourself, video game solutions, social life, and medical recommendations. This corresponds to 1641 instructions over 4560 sequences potential instructions and 511 titles. The total number of words is 61159, this not very large, but we feel sufficient for an indicative evaluation, giving us directions to improve the system. Evaluators had to indicate whether a sequence is:

1. a title,
2. an instruction, with the possibility to give certainty of judgement on 2 values.

The total work took about 15 hours of manual work. Decisions were quite often difficult to make for some types of texts where quite a lot of knowledge of the domain is required, as for video games. Kappa measures were carried out to evaluate agreement and give an indication of the complexity of the tasks. In terms of inter-annotator agreements, we got for instructions, per domain: coocking recipes (0.82), do it yourself (0.76), social life (0.71), video games (0.45) and medical recommendations (0.42). This gives an idea of the complexity of the task (and therefore modulates the results) and of the uncertainty of some measures. Then the two annotators had discussions (about 5 hours) to reach a consensus and propose a unique annotation for all files, and give again a degree of certainty.

The result was then compared to the annotations realized by the programme. These are summarized in the array below for instructions and titles. Our strategy was in general to favor precision over recall, since even if some instructions are not recognized here and there, the question-answering system can still respond accurately. We have not tried at this level to implement an efficient system, however, we can fully parse 50 Mo of web pages in 8.1 seconds, on a pentium3 3GhZ machine with 4 Go RAM.

Title recognition:

domain	recall	precision	kappa
Coocking recipes	0.72	1	0.79
Do it Yourself	0.80	0.96	0.91
Social life	0.69	0.97	0.75
Video games	0.61	0.93	0.77
Medical notices	0.58	0.81	0.89

Instruction recognition:

domain	recall	precision	kappa	certainty
Coocking receipes	0.81	1	0.82	0.88
Do it Yourself	0.77	0.95	0.76	0.84
Social life	0.63	0.94	0.58	0.78
Video games	0.38	0.96	0.48	0.58
Medical notices	0.33	0.95	0.6	0.57

The first three domains give quite good results, while for the last two, results are less good. This is mainmly due to the fact that we designed patterns from prototypical procedural texts, like coocking recipes. Obviously texts like video game solutions have a more unexpected form that would require the development of specific patterns, a solution that we foresee. As can be noted, title recognition gives sligthly better results.

Finally, for instructional compounds, for the three best domains, and with respect to the results obtained in each of these domains, we have the following results, based on a small corpus of data, due to the complexity of the manual analysis:

domain	recall	precision
Coocking recipes	0.95	1
Do it Yourself	0.89	0.98
Social life	0.88	0.98

4.2. Discussion

A point which is worth noticing is that the identification of instructions can be a challenging task for humans too. Apart from domains which mainly use clearly injonctive patterns (cooking recipes, do-it-yourself), instructions can be expressed in a variety of means which are by far less straightforward than imperative or infinive forms. These ambiguous formulations are, for example, the use of passive voice (i), middle reflexives (ii), future tense (iii):

- Les bonnes manières sont inculquées dès le plus jeune âge.
- Les bonnes manières s'inculquent dès le plus jeune âge.
- On inculquera les bonnes manières dès le plus jeune âge.[1]

In the context of a procedural text, a difficulty is that formulations may range from injunctive forms to more neutral statements (X is realized by doing...). Therefore, it is not trivial to identify instructions among other statements such as advices.

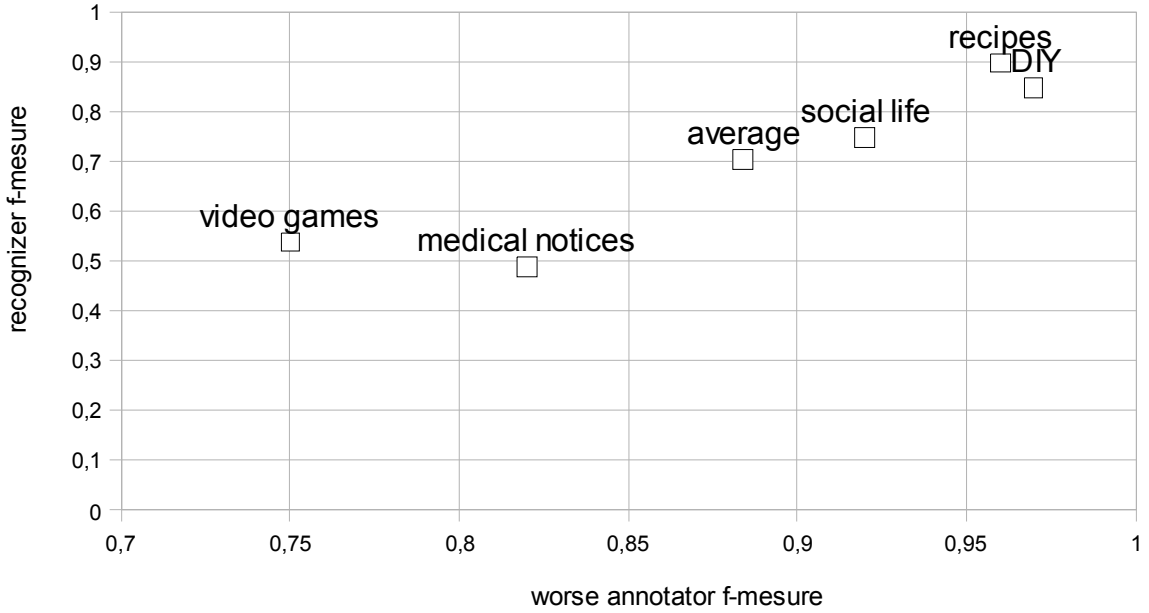
Even more ambiguous is the use of implicit/open-to-interpretation formulations such as: *Le diagnostic de cancer de la prostate repose sur un examen histologique, préalable indispensable à toute décision de traitement.* (~Prostatis cancer diagnosis is based on an histologic examination, which is preliminary to any treatment decision).

The kappa-coefficient and the degree of certainty give an insight about the more or less ambiguous nature of instructions. What is important to notice is that the results of the recognizer decreases quite proportionnally to the results of the worse annotator (evaluated against the common annotation) – except for the video games domain on which the recognizer performs relatively well despite the ambiguity of the instructions:

[1] (i) Good manners are taught at an early age

(ii) no middle-reflexive equivalence

(ii) Good manners will be taught at an early age.



This is the main explanation for the lower rates in the video games / medical notices / social life domains. For example, implicit/open-to-interpretation formulations account for 0,4% of silence in medical notices.

Other causes of silence are, for instance, the taking into account of instructions that spread over several adjoining sequences (0,34% of video games silences; 0,18% in medical notices):

(seq1) Il vous suffit ensuite de créer ce sort:[2]
(seq2) 1. charme 1 pt 5 sec sur contact
(seq3) 2. fortification magie 100 pts pour 20 sec sur soit

Regarding titles, most errors are caused by short sequences bearing no emphasis which induce both noise and silence. In silence cases, these are just titles bearing no emphasis; noise cases are small sequences such as picture caption or any short sequence between two long paragraphs of text mistakenly interpreted as a title.

5. A Few Perspectives

This work is still under research. However, the linguistic structure of texts and the methods to recognize titles, instructions and instructional compounds and the global text structure seem to be on the right track. We obviously need to deepen evaluation for compounds as well as for whole texts, but this is much more difficult due to the complexity of annotations.

To improve the domains with low level results, one direction would be to design dedicated recognizers, with specific patterns. Some more efforts are also necessary in large texts to identify title hierarchies. At the moment, we do not see any simple solution which does not involve pragmatic or domain factors.

The last step of the project is to explore how How-to questions can match with titles (goals), and what kind of results must be returned to the user (the instructions below the title, more data containing prerequisites, several documents, etc.). This project being an ANR-RNTL

[2] All you need is to create this spell :

1. charm 1pt for 5secs on touch
2. fortify magicka 100 pts for 20 secs on self

project, the perspectives include the development of a industrial system and its integration into these partners systems.

Acknowledgements

This project is supported by the ANR under the RNTL programme, project TextCoop, we thank the ANR and our partners: Syllabs, Sinequa and Université de Paris Nord.

References

Aouladomar F., Saint-Dizier P. (2005). An Exploration of the Diversity of Natural Argumentation in Instructional Texts. *5th International Workshop on Computational Models of Natural Argument*, IJCAI. Edinburgh.

Delin J., Hartley A., Paris C., Scott D., Vander Linden K. (1994). Expressing Procedural Relationships in Multilingual Instructions. In *Proceedings of the Seventh International Workshop on Natural Language Generation*, pp. 61-70. Maine, USA.

Delpech E., Murguia E., Saint-Dizier P. (2007). *A Two-Level Strategy for Parsing Procedural Texts*, VSST07. Marrakech.

Kosseim L., Lapalme G. (2000). *Choosing Rhetorical Structures to Plan Instructional Texts*. Computational Intelligence, Blackwell, Boston.

De Rijke M. (2005). Question Answering: What's Next. *The Sixth International Workshop on Computational Semantics.* Tilburg.

Luc C., Mojahid M., Virbel J., Garcia-Debanc C., Pery-Woodley M.-P. (1999). A Linguistic Approach to Some Parameters of Layout: A study of enumerations. In R. Power and D. Scott (Eds.), *Using Layout for the Generation, Understanding or Retrieval of Documents*, AAAI 1999 Fall Symposium, pp. 20-29.

Maybury M. (2004). *New Directions in Question Answering*. The MIT Press, Menlo Park.

Moldovan D., Harabagiu S., Pasca M., Milhacea R., Goodrum R., Gîrju R., Rus V. (2000). The Structure and Performance of an Open-Domain Question Answering System. In *Proceedings of the 38th Meeting of the Association for Computational Linguistics (ACL)*. Hong Kong.

Takechi M., Tokunaga T., Matsumoto Y., Tanaka H. (2003). Feature Selection in Categorizing Procedural Expressions. *The Sixth International Workshop on Information Retrieval with Asian Languages* (IRAL2003), pp.49-56.

Vander Linden K. (1993). *Speaking of Actions Choosing Rhetorical Status and Grammatical Form in Instructional Text Generation,* PhD dissertation, University of Colorado.