Investigating the Structure of Procedural Texts for Answering How-to Questions

Estelle Delpech, Patrick Saint Dizier

Investigating the Structure of Procedural Texts for Answering How-to Questions

Estelle Delpech, Patrick Saint-Dizier
IRIT-CNRS
118 route de Narbonne
31062 Toulouse cedex France
stdizier@irit.fr

October 17, 2007

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.

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 parsing 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).
- 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 different parameters:
  - deontic: obligatory, optional, forbidden or impossible, alternates (or),
  - illocutionary and related aspects: stresses on actions: necessary, advised, recommended, to be avoided, etc.
- The **conditional structure**: introduces conditions over instructions within the compound or even over the whole instructional compound.
- 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: causality, 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.

Let us now give an illustrative example (translated from French), extracted from the 'Do-It-Yourself Home' domain: *In the 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.

3. Recognizing Titles, Instructions and Instructional Compounds

Cleaning Web texts: our system has Web pages as inputs. For an optimal parse, the first task is to remove a number of useless elements (tags and noise like advertising, chats or navigation links), while keeping tags and text portions which are a priori of interest. This is not an easy task, (results under submission). The evaluation of this procedure, designed to be fast and simple, carried out over 100 texts from 8 different domains gives a precision of 0.78, a recall of 0.90 and an f-measure of 0.83. These results are satisfactory w.r.t. our purpose.

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 $< h. >$ 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...). Low level titles even have more unexpected encodings. Encodings may be quite homogeneous within a given web site, but heterogeneity prevails over different sites, even in the same domain.

To recognize titles, we first made a simple selection that consists in keeping all those sequences in a text between $< b >$ or $< h >$ tags which are below 6 words long. Then we applied two selectional criteria: positional: immediate precedence of a set of instructions (when these are recognized) and contents: similarity of the terms in the title with respect to those most frequently used in the paragraph that follows (thematic cohesion). This measures the lexical similarity between an assumed title and the paragraph that follows. Our results show that this similarity is 6 times higher for titles than for sequences in bold which are not titles. We however observed a quite high standard deviation. Final evaluation is given in the section below.

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 95% of our texts not more than 2 levels of titles. While level 1 is often well delimited from the text part, level 2 is often closely associated, as e.g. a bold sequence followed by a semi-colon, or a short spacing. To answer How-to questions, it seems

that only level 0 and 1 titles are relevant. One remaining difficulty is that titles have often a very elliptic structure.

Recognizing instructions and instructional compounds: 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. Verbs must be action verbs (this may depend on the domain and subsets can be defined for each domain to improve accuracy). They must have in French specific forms, in decreasing frequency order: imperative, infinitive, modal + infinitive, dummy pronoun 'on' + finite verb, middle reflexive constructions, and gerundive forms. The frequency usage of each of these forms largely varies across domains (e.g. coocking receipes 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 25 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. $< li >$ 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.

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.

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, medical recommendations. This corresponds to 1641 instructions over 4560 sequences potential instructions and 511 titles. 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:

- a title,

- an instruction, with the possibility to give certainty of judgement on 3 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 have a measure of the complexity of the tasks. In terms of inter-annotator agreements, we got for instructions, per domain: coocking recipes (82%), do it yourself (76%), social life (71%), video games (45%) and medical recommendations (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.

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 1 Mo of web pages in 7.25 seconds, on a pentium3 3GhZ machine with 4 Go RAM.

Instructions recognition:

domain	recall	precision	F-measure
coocking receipes	0.81	1	0.89
do it Yourself	0.77	0.95	0.85
social life	0.63	0.94	0.75
video games	0.38	0.96	0.54
medical notices	0.33	0.95	0.49

Titles recognition:

domain	recall	precision	F-measure
coocking receipes	0.72	1	0.83
do it Yourself	0.8	0.96	0.87
social life	0.69	0.97	0.80
video games	0.61	0.93	0.74
medical notices	0.58	0.81	0.67

The first three domains give quite good results, while for the last two, the poor quality of texts and their high diversity explains the moderately good results, which was expected. As can be noted, title recognition gives sligthly better results.

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.).

6. References

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

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

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

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

De Rijke, M., *Question Answering: What's Next?*, the Sixth International Workshop on Computational Semantics, Tilburg, 2005.

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

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