Challenges of Discourse Processing

Challenges of Discourse Processing: The Case of Technical Documents

Edited by

Patrick Saint-Dizier

CAMBRIDGE SCHOLARS PUBLISHING

Challenges of Discourse Processing:
The Case of Technical Documents,
Edited by Patrick Saint-Dizier

This book first published 2014

Cambridge Scholars Publishing

12 Back Chapman Street, Newcastle upon Tyne, NE6 2XX, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN (10): 1-4438-5583-9, ISBN (13): 978-1-4438-5583-9

TABLE OF CONTENTS

LIST OF TABLES

PREFACE

The production of technical documents of various kinds is often considered in most industrial sectors as a costly and painful activity. Technical documents mainly include procedural texts (e.g. for maintenance, production launch), equipment and product manuals, various notices such as security notices, regulations (security, management), repositories of requirements and business rules (e.g. concerning the design properties of a product) and product or process specifications. Technical documents may be very large, with a complex conceptual structure; they often cover areas which are sensitive in terms of safety and security.

About two decades ago, technical documents have motivated a number of language processing research activities, with the development of authoring recommendations. In linguistics, the technical document genre was investigated in order to outline its language but also its cognitive and ergonomics properties. The new types of technical documents (e.g. requirements and business rules), the increasing constraints on technical documentation production and control (e.g. certifications, safety and security norms), and the emergence of more advanced needs (e.g. more accurate authoring recommendations, coherence controls between documents, traceability and update) require the development of a new generation of linguistic analysis and authoring tools. This is now made possible in particular thanks to the major developments of natural language processing methods, technologies, and resources over the last decade.

Discourse analysis still remains at the level of theoretical considerations with little application capabilities. The development of discourse annotation tools and methods may in the future allow the development of discourse parsers working with a good accuracy on a large diversity of types of documents. However, discourse analysis remains a very difficult and challenging area in natural language processing due to the complexity of discourse constructions whose analysis often involves language associated with domain knowledge and reasoning.

Technical documents are relatively constrained in terms of language diversity and complexity: the goal is to make sure that users can understand these documents without any ambiguity and therefore without any risk for their health or for the environment. Technical documents tend

to minimize the distance between language and action: strategies to realize a goal are made as immediate and explicit as possible.

These considerations make it possible to develop an accurate discourse analysis of technical documents which can be used to improve their overall quality. From a more theoretical point of view, technical documents form a very interesting corpus of texts for discourse analysis. They allow the development of theoretical elements and tools for explanation and argumentation, two major aspects of discourse, which are crucial in technical documentation.

In this book, we show that linguistic analysis and natural language processing methods can efficiently and accurately be used to represent the discourse structure of technical documents and to improve their form and contents, independently of the industrial sector and activity. This book aims at presenting well-founded and concrete solutions, which can be deployed in industrial contexts.

The research from which this book emerged was funded by the French *Agence Nationale pour la Recherche* (ANR) from 2005 till 2013, via the TextCoop and then the Lelie projects. These two projects have motivated a large number of interactions with linguists, technical writers and researchers from the industry and academic circles involved in linguistics, computer science, document production and cognitive ergonomics. Collaborations in these various disciplines have allowed the development of useful principles and prototypes which are now been tested and customized in various industrial sectors. I thank our funding agency, the ANR, and my institution, the CNRS, for their continuous support. This book has also benefited from the work and the interactions with a number of researchers, users, technical staff and students. In particular, my special thanks to *Camille Albert, Farida Aouladomar, Florence Bannay, Flore Barcellini, Sarah Bourse, Estelle Delpech, Lionel Fontan, Marie Garnier, Corinne Grosse, Mathilde Janier, Juyeon Kang, Naima Khoujane and Marie-Christine Lagasquié.* I also want to thank the companies that helped us in our investigations, in particular: EDF, SNCF, EADS, Liebherr, Orange, and Thomson-Reuters.

CHAPTER ONE

AN INTRODUCTION TO THE STRUCTURE OF TECHNICAL DOCUMENTS

PATRICK SAINT-DIZIER

Introduction

Technical documents form a linguistic genre with very specific linguistic constraints in terms of lexical realizations and grammatical and style constructions. Typography and overall document organization are also specific to this genre. Technical documents cover a large variety of types of documents: procedures (also called instructional texts), which are probably the most frequently encountered in this genre, equipment and product manuals, various notices such as security notices, regulations (security, management), requirements (e.g. describing the design properties of a product) and product or process specifications. These documents are designed to be easy to read and as efficient and unambiguous as possible for their readers. For that purpose, these documents tend to follow relatively strict authoring principles concerning both their form and contents. However, depending in particular on the industrial domain, the required security level, and the target user, major differences in the writing and overall document organization quality can be observed.

An important feature is that technical documents tend to minimize the distance between language and action. Instructions 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, and therefore potential hesitations, errors or misunderstandings. Technical documents are thus oriented towards action, they often combine instructions with icons, images, graphics, summaries, etc.

In this book, we mainly consider technical documents produced by the industry for their staff or for their industrial customers. Technical documents designed for the large public follow the same principles, but in

a more shallow way. Procedures are documents designed to realize a certain task, possibly decomposed into subtasks. There are several types of tasks which can be described by procedures such as equipment installation or maintenance, production activities, and staff management. The size of a procedure ranges from one page for short notices to 200 pages. Procedures are written using various types of text editors, the most common one being Microsoft Word. Some large companies have their own authoring environment, in general based on XML. In that case, large hierarchies of XML documents may be produced and stored in text databases. The end-user gets in general a PDF document adapted to his task, or documents readable on tablets or on any equivalent electronic device, with the possibility to get additional information in an interactive way. Vocal applications also exist and are used e.g. when there is no possibility for operators to use a written support.

There is a large diversity of procedural texts, with different objectives and styles. In most cases, industrial procedures are rather uniformly composed of a hierarchy of titles that reflect a task-subtask hierarchy and instructions that describe how to realize each task or subtask. Titles and instructions form the backbone of procedures. A number of additional elements are often associated with instructions or subtasks in order to guide, help or warn the user in particular when there are difficulties or risks. We call these elements the *explanation structure of a procedure*. As will be seen in the next chapter, the explanation structure plays a major role in the correct realization of a task. The explanation structure includes, among others, definitions, illustrations, reformulations or elaborations, and also arguments such as warnings and advice. Explanation is aimed at providing users with information that may possibly partly contradict their beliefs or assumptions. Arguments do not provide any new information, they are designed to convince the user to realize the task as required.

Equipment and product manuals describe the structure or the composition and the properties of the equipment or product at stake, the precautions to take when using it, possibly how to store, install and maintain it. These documents have a rich structure, made of definitions and schemas, instructions, explanation and arguments.

Regulations and requirements form a specific subgenre in technical documents: they do not describe how to realize a task but the constraints that hold on certain types of tasks or products (e.g. safety regulations, management and financial regulations). Similarly, process or product specifications describes the properties or the expectations related to a product or a process which is being elaborated. These may be used as a

contractual basis in the realization of the product at stake. Specifications often have the form of arguments.

Technical documents are seldom written from scratch: they are often the revision, the adaptation, or the compilation of previously written documents. Technical documents of a certain size are not produced by a single author: they result from the collaboration of several technical writers, technicians and validators. Their production may take several months, with several cycles of revisions, validation and updates e.g. coming from returns of experience from users. Their life cycle ranges in general between two and ten years. Text revision and update is a major and complex activity in industrial documentation, which requires very accurate authoring principles, text analysis and validation cycles to avoid any form of "textual chaos".

Elaborating a new technical document from previously existing ones is a major source of errors in spite of several proof-reading steps. These errors may result from incorrect updates, missing information, incorrect references, incorrect copy-paste operations, or from recent security norms which are not fully adapted to the situation described in the document. Therefore, technicians in operation or users must manage these deficiencies, e.g. finding errors and gaps, and elaborating by themselves alternative solutions. This is obviously a major source of stress, risk and accident. Controlling the quality of a document, in terms of form (from typography and general lay-out to style) and contents, is therefore essential to prevent risks, production errors and customer dissatisfaction. Quality analysis is also crucial to make sure regulations are enforced, or to get a certification or an adequate level of insurance.

In this book we show that linguistic analysis and natural language processing methods and tools can efficiently be used to analyse the form and contents of technical documents. Besides linguistic observations, this book aims at presenting concrete solutions for improving the form and contents of technical documents based on advanced, logic-based natural language processing techniques. This book focuses in particular on conceptual and discourse structure analysis within the context of technical documents, which is more restricted than in general language. Concrete solutions are addressed in depth in this book from a number of views that complement each other:

- First, some introductory considerations are presented on the typology of technical documents, on the way technical documents should be produced in interaction between authors and users, and on their global conceptual and discourse structure (Chapter 1, in the sections that follow).

- Then, the way a technical document is organized from a conceptual and a linguistic point of view is investigated: its discourse structure and in particular the title-instruction organization and also its explanation and argumentation structures since most technical documents offer a rich user support (Chapter 2).

-Next, the authoring recommendations that technical writers should follow in a number of companies are surveyed. Recommendations concern the uses of business terms, lexical choice, grammatical constructions and style (Chapter 3). These recommendations also reflect the principles developed for Controlled Languages.

-Then, Chapter 4 introduces the programming language Dislog that runs on a dedicated platform, <TextCoop>. Dislog is primarily designed for discourse processing. The discourse structures presented in Chapter 2 can be processed by this environment.

-The way to use this platform is explained in detail in Chapter 5, which is a kind of user manual.

-Chapter 6 is devoted to the linguistic structure of requirements and specification documents. These types of documents have a specific discourse structure which is outlined.

-Finally, in Chapter 7, the Lelie project is introduced. This project integrates most of the aspects presented above. The main aim of Lelie is risk prevention from an accurate analysis of the quality of written technical documents, in terms of form and contents. Risks include health as well as ecology or finance.

On the basis of a linguistic analysis that includes the lexical, grammatical, discourse and style levels and the use of natural language processing tools, this book aims at providing some essential technical answers to the following major challenges of technical documentation production:

- How to make sure that a technical document is written clearly and comprehensively enough to avoid any major risks (e.g. installation errors, task misconceptions, etc.)?
- How to make sure that technical documents can be accepted and used by technicians or end-users? This means that these documents are well understood, feasible, accepted, e.g. in terms of complexity and workload, and without any useless consideration.
- How to make sure that a technical document, and a procedure in particular, has a good internal coherence? This means that the instructions which are given allow an adequate realization of the goal(s) of the procedure.

-A last challenge is how to make sure that a procedure, via its instructions, follows the various public and business regulations it is concerned with?

Some Considerations on the Typology of Technical Documents

A technical document is a coherent set of statements whose goal is to reach a certain goal (for procedures and security requirements), to describe the properties and the way to use an equipment or a product, or to make more explicit the existing or expected properties of a system (e.g. using requirements) (Adam 1987). Technical texts may have different aims depending on their type, among which:

- ***Texts implementing regulations*** have a very efficient and injunctive form. A typical example are security notices (e.g. for fire prevention): these documents are very short and serve a very precise purpose. They describe in a very clear way the behaviour of the reader when the situation at stake occurs. The style is simple, direct and very injunctive.
- ***Texts having a "programmatic" nature:*** their goal is mainly to offer a structured knowledge on a given topic: use of an equipment, development of a know-how, or description of the elements to prepare, use or elaborate in a given situation. The form of these texts can be made quite flexible and attractive to the reader. Various forms of illustrations are frequently used.
- ***Procedural texts***: describe how to realize a precise action by means of temporally ordered instructions. Large companies have in general their own authoring recommendations that technical writers must follow.
- ***Texts describing the design or the functional properties*** of systems or equipment (design requirements) (Hull et al. 2011). Their style follows very precise guidelines, sentences are short and as unambiguous as possible since it is crucial to avoid any type of misunderstandings that would lead to incorrect products or product uses.
- ***Advice texts:*** are less strictly written and organized (no temporal structure a priori). These include texts devoted e.g. to staff management, meeting organization, social behaviour in companies or with customers.

Elements of an Analysis of the Gaps between Technical Writers and Operators

When investigating the form of technical documents, a major point is to observe how technical writers proceed when they elaborate a document. This means investigating technical document validation steps, how their life cycles are implemented, and possibly how returns of experience are taken into account. In document production, the emphasis is more frequently put on the producer than on the consumer, while the latter should in fact be the most important parameter of the process. We present here a short analysis of the gaps that often exist between technical writers and technicians in operation. These latter often have to find solutions to errors or gaps they find in technical documents. This is obviously a stressful situation that may lead to accidents. In the maintenance domain, about 78% of the technicians recognize that they found errors in procedures and that they had to elaborate alternative solutions without any external support. This clearly shows the difficulty of the task since even experienced technical writers encounter difficulties to produce stable and comprehensive documents. It is interesting to note that a number of technical writers have a strong experience as operators.

A number of investigations in cognitive ergonomics, based on precise protocols, have analysed problems of adequacy and efficiency of written procedural texts. The analysis was realized on the global structure of a procedure as well as on the structure of every instruction (e.g. (Schriver 1989)). The main reasons of a lack of adequacy and efficiency can be summarized as follows:

- A partial incompatibility between the document structure and the mental model of the operator who has to convert written instructions into actions. For example, an instruction may need to be decomposed into several concrete actions or, vive-versa, a set of instructions needs to be grouped to form a single action. The temporal organization of instructions may also be somewhat adapted to a given situation by the operator.
- An insufficient consideration for human factors (such as complexity of an action, communication in a noisy environment, isolated worker situation) that generates stress and psychological and social problems. Similarly, an inadequate or insufficient analysis of the various returns of experience produced by operators generates frustration.

- The difficulty and the stress associated with unexpected situations that operators often have to manage alone and in emergency situations.

These elements are essential and must accurately be taken into account when producing technical documents. However, besides authoring guidelines that need to be improved, user reactions must be taken into account and anticipated.

Towards the Elaboration of Quality Criteria for Technical Documents

In Chapter 3 of this book, we present criteria to improve the quality and ease of use of technical documents. These criteria are elaborated from the observation of technical writers at work, e.g. how they interact between each other and with technicians, and then how technicians read and analyse the technical documents that they use. These observations are paired with recommendations produced within the framework of ***Controlled languages*** (e.g. (Weiss 1991), (Alred 2012), (O'Brien 2003), (Wyner et al. 2002), see also internet links in the bibliography section). Note that controlled language guidelines operate only at the "surface" level of documents, not at the contents level.

Another difficulty is that, quite often, large documents are produced by teams of technical writers, who spend a lot of time revising text portions written by their colleagues. Revising such documents raises problems of competence and responsibility. Finally, when writing guidelines are very strict, it turns out that they are not strictly followed in a number of cases. On the other hand, when they are too shallow, they are not taken into account seriously. Therefore, there if a tradeoff to find between these two extremes when developing guidelines.

Technical documentation production can be realized according to two very different perspectives: (1) either the technical writer is strongly guided and has little choice in the constructions and terms he can use, this is the language technology proposed in the *boilerplates* perspective or (2) he can write documents with a certain freedom, following authoring principles, and with an *a posteriori* control on what he has written. The first perspective is relevant e.g. for requirements (Chapter 6). Boilerplates are indeed used for producing very simple documents such as security notices or for producing software requirements, which must follow extremely simple structures. The boilerplate technology is not adapted to the production of large and often complex procedures, where a certain

flexibility in language must remain the rule. In that case *a posteriori* controls are much more adapted and better accepted by authors.

Let us review here a few general principles concerning document production quality criteria that we have elaborated from the literature and our observations, these are further developed and illustrated in Chapter 3. The motivation for high quality documents is often to limit the cognitive load imposed to the operator when he reads the document. The main criteria are:

- ***Statements simplicity:*** clear, precise and non-ambiguous terms in the domain considered must be used with a syntax as elementary as possible. In an instruction, for example, the verb *count* will be preferred to a verb such as *observe* since this latter verb does not correspond a priori to a precise action. Lists of preferred terms are often given in company's guidelines. Statements are preferably in the active voice, with the verb complements in their canonical order (i.e. as expected by the verb "base" form).
- ***Statements conciseness***: useless words or constructions (modals, auxiliaries) must be avoided, long expressions must be replaced by shorter ones when the meaning is not affected.
- ***Document cohesion***: the design of the document, as well as its style and grammatical constructions must be used in a homogeneous way. Links between paragraphs must be clear and stable. Tables, legends and graphics must be organized in similar ways. Abbreviations, units and other such elements must always be used in the same way, with no variants.
- ***Clarity of the main elements***: the main elements, instructions, goals, warnings, etc. should be the most visible, possibly with an adequate typography. Words which are too general have in general a relatively empty or vague meaning (*impact, interface*). These words must be avoided. Similarly, pronouns with unclear references or fuzzy terms must not be used.
- ***Accessibility and readability*** of the document: the text must be easy to read and to understand by the target user, using words and constructions he masters.

These principles apply to any type of technical document. Product specifications and safety and design requirements may be associated with additional recommendations which are proper to the activity they are related to.

Some Considerations on the Conceptual and Discourse Structures of Technical Documents

A few Conceptual Principles

Technical documents are specific forms of documents, satisfying constraints of economy of means, expressivity, precision and relevance. They are in general based on a specific logic and type of organization, made up of presuppositions, implicit goals, causes and consequences, inductions, warnings, anaphoric networks, etc. They also include more psychological elements (e.g. *to stimulate a user*). The goal is to optimize the logical sequencing of instructions and make the user feel safe and confident with respect to the goal(s) he has to achieve (e.g. *clean an oil filter, learn how to organize a customer meeting*).

Procedural texts, from this point of view, can be analyzed not only as sequences of mere instructions, but as efficient, one-way (i.e. no contradiction, no negotiation) argumentative discourses, designed to explain to a user how to realize a task, providing motivations and helping him to make the best decisions at each step of his work. This type of discourse contains a number of facets, which are often associated in a certain way with explanation. Procedural discourse is indeed informative, explicative, descriptive, injunctive and sometimes narrative and figurative. These features are used with large variations depending on the author, the activity and the target reader. From that point of view, given a certain goal, it is of much interest to compare or contrast the means used by different authors, possibly for different audiences.

Writing a procedure and producing explanations is a rather synthetic activity whose goal is to use the elements introduced by knowledge elicitation mechanisms to help the user to acquire additional knowledge or revise his skills, beliefs and knowledge when these are not fully correct or up to date. Explanation may also induce generalizations, subsumptions, deductions, and the discovery of relations between objects or activities and the goals to reach. Argumentation does not a priori entail the acquisition of new information: it is designed to convince the reader that the procedure as it is written is among the best ways to realize it in a safe, efficient and economical way.

The authors of technical documents must consider three dimensions (Donin et al., 1992), (Van der Linden 1993) when producing such a document:

(1) ***cognitive***: it is essential to make sure that notions referred to are well-mastered and understood by the target users,

(2) ***epistemic***: it is crucial to take into account, possibly to deny them, the beliefs of those users, and
(3) ***linguistic***: it is necessary to use an appropriate language. This means, as already advocated, to adjust to the target user the accuracy of words, the technical level, the complexity of sentences and paragraphs, and the visual and typographic structure of the text.

The authors of technical documents start from a number of assumptions or presuppositions about potential users, about their knowledge, abilities and skills, but also about their beliefs, preferences, opinions, ability to generalize and adapt a situation, perception of generic situations, and ability to follow discursive processes. For example, users must often adapt instructions to their own situation, which is not exactly the situation described in the procedure. The producers of procedural texts have, from this basis, to re-enforce or weaken presuppositions, to specify some extra knowledge and know-how, and possibly to outline incorrect beliefs and opinions. They have to convince the reader that his text will certainly lead to a successful realization of the task at stake, modulo the restrictions they state (Aouladomar 2005).

Technical documents are in general highly structured and modular. They exhibit a particularly rich micro-rhetorical structure integrated into the syntactic-semantic structures of instructions. Procedural texts are a particularly difficult exercise to realize. For example they must make linear, because of language constraints, actions which may have a more complex temporal or causal structure. Connectors and referents contribute to implement this linearity, making the task simpler. Texts are also expected to be locally and globally coherent, with no contradictions, and no space for hesitation or negotiation.

Another important feature, which is rather implicit, is the way instructions or groups of instructions are organized and follow each other, and how the logic (objective aspect) and the connotations (subjective aspects) that underlie this organization (sequential, parallel, concurrent, conditional, multi-user, etc.) cooperate to produce optimal documents. The general organization of a document, its lay-out and its typography contribute to optimize its use. Going one step further, the relation between text and pictures, diagrams or images, which is not addressed in this book, is an important issue in technical documents. Pictures and diagrams complement or illustrate the text, they facilitate understanding via a visual support but may also raise ambiguities.

Finally, there is most of the time no syntactic sign characterizing the author or the user: there is no use of personal pronoun like "You" or "We". However, the author is implicit in languages such as French or Spanish by

the use of imperative or infinitive verbal forms. The most common form used by authors to express instructions is the injunctive discourse. It characterizes several modalities of discourse: orders, preventions, warnings, avoidances, advice. These all have a strong volitional and deontic dimension. Injunctive discourse shows how the author of an instructional text imposes his point of view to the user: instructional texts are a prototypical example of a logic of action. Injunction is particularly frequent in security notices or in any difficult task where security is important. The strength of an injunctive statement is measured via the illocutionary force of the statement, often realized by verbs combined with adverbs.

A Global Analysis of the Structure of Procedures

The structure of most technical documents is highly hierarchical and very modular. These documents often follow authoring and organization principles proper to a company or to an organization that specify how to organize the different levels of the document.

The higher level of technical documents often starts by a few introductory words defining the task described in the document. These words are associated with general considerations such as purpose or motivations, scope, limitations and context. This higher level also contains a date of issue, a table of contents, a list of contributors and indications about revisions carried out, its maturity status and its distribution or confidentiality level. It can then be followed by definitions, examples, scenarios or schemas. A technical document often ends by annexes and a set of references, e.g. to previous documents or to external documents where more information can be found about a precise topic or equipment.

Lower levels include a hierarchy of sections that address the different facets of the goals the document proposes to achieve. Each section may include for its own purpose general elements similar to those given above followed by the relevant instructions. Instructions can just be listed in an appropriate temporal order or be preceded by comments. Each instruction or group of instructions can be associated with e.g. conditions or warnings, goal expressions and forms of explanation such as justifications, reformulations or illustrations.

The temporal aspects in a procedure are important. In general, to facilitate their execution, instructions describe actions that strictly follow each other, in a very clear linear order. However, there are cases that involve partial or total instruction overlap. Some actions are also triggered by others or must show a certain degree of synchronization; they must

therefore be explicitly related by an appropriate connector (e.g. *heat the probe till it reaches 80 degrees and then stop the electricity*). Except for these latter cases where it is clearly stated, temporal connectors are often not explicit. They are often realized by punctuation marks, the ambiguous "*and*" or the simple succession of instructions.

Depending on the complexity of the task to realize, the risks and the technical level of the operator, instructions may be associated with more or less textual information. Instructions may also be marked as optional or they may need to be repeated till a certain condition is reached.

Each instruction in a procedure may be realized as a sentence or, equivalently, as an element of an enumeration. Quite frequently, instructions are grouped by small units which share a number of parameters (objects, conditions, goals, etc.). These instructions are realized in a single sentence with the use of referents or ellipsis (e.g. *before you start, you must open the box, clean its interior and plug in the white cable).* We call this small group of instructions an instructional compound, which is often considered as a single operational unit.

Considering the structure of instructions, two main works introduce the major foundational aspects of instruction contents. These will be used as the starting point of the development of the discursive structure of procedural texts that we have elaborated and which is developed in Chapter 2. First, (Bieger et al., 1984-85), based on a cognitive approach, propose a taxonomy of the contents of instructions organized in nine points:

-inventory (objects and concepts used),
-description or definition (of objects and concepts),
-operational (information that suggest the agent how to realize an action) such as manners or means to use,
-spatial (spatial data about the actions, how they follow each other or how they are combined in a spatial environment),
-contextual aspects,
-covariance (of actions, which must evolve in conjunction),
-temporal aspects,
-qualificative (limits of an information), and
-emphatic (redirects attention to another action).

The second investigation we consider was realized within the framework of Computational Linguistics, it is due to (Kosseim, 2000). She isolated from corpus analysis nine main structures or operations, called *semantic elements*. These partly overlap or complement the previous ones, with a more operational view of procedures:

-sequential operations: a necessary structured group of actions that the agent must realize,
-object attribute: description meant to help understand the action to realize, with what means or objects,
-material conditions: concrete environment in which an action must be carried out,
-effects: consequences of the realization of a group of operations on the world (or environment of the user),
-influences: explain why and how an operation must be realized and its consequences on others,
-co-temporal operations: express operation synchronization, overlap, or any other temporal organization,
-options: optional operations, designed to improve the quality of the results, these are called advice in our terminology. Optionality may receive several degrees.
-preventions: describe actions to avoid, these are called warnings in our terminology,
-possible operations: possible operations to do in the future, such as evaluations or controls.

As the reader may note it, the internal structure of instructions may be quite complex. An instruction is not just an action composed of a main verb, an object and an instrument, but it is often the conjunction of many conceptual elements which are necessary for a correct execution of the action(s). A number of these elements are formalized in the next chapter, Chapter 2, where their grammatical and discourse structures are investigated. Principles and recommendations on how to write instructions are given in Chapter 3.

The goal of this introductory chapter was to give an overview of technical documentation production and technical document overall structure. The chapters that follow investigate some of the problems advocated here; they propose solutions based on linguistic analysis and natural language processing tools. These chapters can be read independently.

CHAPTER TWO

THE LANGUAGE OF TECHNICAL DOCUMENTS: A LINGUISTIC DESCRIPTION OF ARGUMENTATION AND EXPLANATION

PATRICK SAINT-DIZIER

Introduction and Context

In this chapter we investigate the linguistic structure of technical documents. We focus in particular on the structure of procedural texts since it is the most central and crucial document in the industry, but the elements presented here are also applicable to the other types of technical documents, such as requirements, which are developed in Chapter 6. As indicated in Chapter 1, procedural texts cover a large diversity of categories of texts, with various styles, e.g. maintenance and installation of equipment, security guides, medical notices, social behavior recommendations, management guides and legal documents. Several features of the present chapter have been initially developed in (Bourse and Saint-Dizier 2012).

Procedural texts consist of a sequence of instructions, designed with some accuracy in order to reach a goal (e.g. *assemble a computer*). Procedural texts often include sub-goals (e.g. *dealing with the different subparts of a computer*). Goals and sub-goals are most of the time linguistically realized by means of a hierarchy of titles and subtitles. The objective is that the user must carefully follow step by step the goal / sub-goals structure and the instructions in order to correctly realize the task at stake. The general conceptual structure of technical documents is presented in Chapter 1.

Analyzing the structure of procedural texts means identifying titles (which convey the main goals of the procedure), sequences of instructions serving these goals, and a number of additional structures such as prerequisites, warnings, advice, illustrations, evaluations, reformulations,

etc. (Van der Linden 1993), (Takechi et ali. 2003), (Adam, 2001). Procedural texts follow a number of structural criteria, whose realization may depend on the author's writing abilities, on the target user, and on traditions or recommendations associated with a given domain. Procedural texts, as introduced in Chapter 1, can be regulatory, procedural, programmatory, prescriptive or injunctive.

Procedural texts are complex structures, they often exhibit quite a complex structure with both rational (the instructions) and "irrational" aspects, mainly composed of advice, warnings, contextual restrictions, conditions, expression of preferences, evaluations, reformulations, user stimulations, etc. These latter form what we call the ***argumentative and explanation structures***; they motivate and justify the goal-instructions structure, viewed as the backbone of procedural texts.

Argumentation includes in particular advice and warnings. Argumentation is very useful, sometimes as important as instructions. Arguments are designed to motivate the user to realize the task as explained, otherwise he may undergo some difficulties or risks. Explanation provides a strong and essential internal coherence to procedural texts: while arguments are designed to convince a user to realize a task are required, explanation provides additional knowledge to the user, possibly in contradiction with his beliefs. Argumentation and explanation complement each other.

From a theoretical point of view, the structure of technical texts may be represented by means of rhetorical relations (Rhetorical structure theory, RST (Mann and Thomson 1988 and 1992)). The relations presented in this chapter are close to RST relations, but they are at the same time simpler and more specific to the technical document genre. In this book, we briefly address RST features in general since they are well developed in the literature (e.g. (Taboada et al. 2006), (Taboada, 2006)).

An important aspect of this chapter is the accurate identification of the explanation structures and the communication goals they serve in procedural texts in order to better understand explanation strategies deployed by technical writers in precise, concrete and operational situations, so that they can be improved and reproduced in other contexts.

The analysis reported in this chapter was carried out on a development corpus of about 200 French and English texts from various industrial sectors, in particular: chemistry, energy, transportation, health, food processing, etc.

This chapter gives some foundations for argumentation and explanation in terms of structure and functions within the framework of technical documents. We first introduce the notion of explanation in

general, as it has been addressed in the literature and then address the current discourse analysis challenges since argumentation and explanation are very much related to discourse. Argumentation being a complex area, we present in this book its uses in technical texts. In a second stage, we introduce the functions that explanation plays in technical documents, which is much more restricted and controlled than in ordinary circumstances. For that purpose, we introduce the notion of elementary explanation functions, as observed from corpus analysis and common practices given in authoring recommendation manuals. Explanation schemes, which are the language realizations of explanation functions are then introduced. This chapter ends by a relatively detailed analysis of a number of discourse structures which are the most prominent in argumentation and in explanation schemes. These structures are given in a standard grammatical form. Chapter 4 and 5 show how these structures can be used in an automatic analysis of discourse structures. In Chapter 6, these structures will be considered again to account for the structure of requirements, which is a specific activity in technical document production.

Argumentation in Action

Argumentation is found in the description of goals, alternative choice statements, warnings, and within instructions. The four major forms of arguments frequently found in procedures (Aouladomar 2005) are described below. Verb classes referred to are in general those specified in WordNet (Fellbaum 1998). At this stage a few linguistic marks and syntactic schemas are given. These are investigated in more depth at the end of this chapter.

Arguments related to goals are the most frequent ones. They usually motivate a goal implemented as a subtask, a set of instructions or more locally an instruction. They must not be confused with purpose clauses: their role is to justify the action, outlining the importance of that goal, the potential risks and difficulties. Their general syntax is the following:

(1) purpose connectors + infinitive verbs,

(2) causal connectors + deverbal or

(3) titles, with a causal or a purpose connector: *to, for in order to*,

e.g. do X *to remove safely the bearings*, make Y *for a correct cleaning of the universal joint shafts*.

Prevention arguments: are based on a "positive" or a "negative" formulation. Their role is basically to explain and to justify an instruction

or a group of instructions. Negative formulations are easy to identify, they are realized by one of the following syntactic schemas:

(1) negative causal connector + infinitive risk verb,
(2) causal connectors + modal + VP(infinitive verb),
(3) negative causal mark + risk verb,
(4) positive causal connector + VP(negative form),
(5) positive causal connector + prevention verb.

The grammatical and lexical elements in these constructions are in particular:

– negative connectors*: otherwise, under the risk of,* (e.g. *otherwise you may damage the connectors*).
– risk verb class*: risk, damage,* etc. (e.g. *in order not to risk the user's life*).
– prevention verbs: *avoid, prevent*, etc. (e.g. *in order to prevent the card from skipping off its rack*).
– positive causal mark and negative verb form: *in order not to*, (e.g. *in order not to make it too bright, in order not to spoil the fruit*).
– modal SV: *may, could*, (e.g. *because it may be prematurely worn due to the failure of another component*).

Positive formulation marks are the same as for the arguments related to goals described above, with the following syntactic schemas:

(1) purpose mark + infinitive verb,
(2) causal subordination mark + subordinate proposition,
(3) causal mark + proposition,

with:

– purpose marks: *so as to, for*,
– causal marks: *because, this is why*, etc.
– causal subordination marks: *so that, for*.
– the verbs encountered are usually of "conservative" type *: preserve, maintain*, etc. (e.g. *so that the axis is maintained vertical*)

Performing arguments: are less imperative than the previous ones, they express advice or evaluations. Their corresponding syntactic schemas are:

(1) causal connector + performing NP,
(2) causal connector + performing verb,
(3) causal connector + modal-performing verb,
(4) performing proposition.

Resources or structures are, for example:

– performing verbs: *allow, improve*,
– performing NP: (e.g. *for a better performance, for more competitive fares)*

– performing proposition: (e.g. *have small bills, it's easier to tip and to pay your fare that way*).

Threatening arguments: have a strong impact on the user's attention when he realizes the instruction, the risks are made very clear via very injunctive schemas. These arguments follow one of the following syntactic schemas:

(1) otherwise connectors + consequence proposition,

(2) otherwise negative expression + consequence proposition,

with, e.g.:

– otherwise connectors: *otherwise*,

– otherwise negative expression: if … do not … (e.g. *if you do not pay your registration fees within the next two days, we will cancel your application*)

Besides these four main types of arguments, some forms of stimulation-evaluation (*what you only have to do now...*) and evaluation can be found.

Explanation in Action

Explanation and its relations to language, cognition and linguistics is a relatively new and vast area of investigation. Explanation analysis involves the taking into account of a large number of aspects, e.g. typography, syntax, semantics, pragmatics, domain and contextual knowledge, and user profile.

At the moment, explanation is essentially developed in sectors where didactics is involved, e.g. writing recommendations for producing essays or, in interactive environments, in systems such as helpdesks. In artificial intelligence, explanation is often associated with the notion of argumentation (Reed 1998) and (Walton et al. 2008), but argumentation is just one facet of explanation. Let us also note investigations on causality, and an emerging field around negotiation and explanation in multi-agent systems associated with an abstract notion of belief (Amgoud et al. 2001). Two decades ago, simple forms of explanation were used to produce natural language outputs for experts systems, often from predefined templates. The goal was to justify why a certain proof was produced and why a certain solution was proposed as a result of a query. In the same range of ideas, natural language generation planning made some use of explanation structures, e.g. (McKeown 1985).

Interesting principles have emerged from these works which have motivated the emergence of interdisciplinary research. For example, in ergonomics and cognitive science, the ability for humans to integrate explanation about a task described in a document or on an electronic

device (possibly via a guidance system) when they perform that task is investigated and measured in relation with the document properties (Lemarié et al. 2008), (Bieger and Glock 1984). In linguistics, a lot of efforts have been devoted to the definition and the recognition of discourse frames (Webber et al. 1990, 2004) (Miltasaki et al. 2004) (Saito et al. 2006) and the linguistic characterization of rhetorical relations (Mann and Thompson 1988; 1992), (Longacre 1982), which are, for some of them, central to explanation (Rösner and Stede 1992), (Van der Linden 1993). However, we now observe a proliferation of rhetorical relations with various subtleties, which, for some of them, turn out to be quite difficult to recognize solely on the basis of language marks since they involve quite a lot of pragmatic considerations and domain knowledge. Finally, explanation is a field which is investigated in pragmatics, e.g.: cooperativity principles (Grice 1978), dialogue organization principles and speech acts, e.g. (Searle et al. 1985), explanation and justification theory (Pollock 1974) and in philosophy, e.g.: rationality and explanation, phenomenology of explanation, causality (Keil and Wilson 2000), (Wright 2004) and, with a perspective on action, (Davidson 1980).

Explanation is in general structured with the aim of reaching a goal. This goal may be practical (e.g. how to reach a certain location) or more interpersonal or epistemic (e.g. convince someone to do something in a certain way, negotiate with someone while providing explanation about one's point of view). Explanation is in fact often associated with a kind of instructional style, explicit or implicit, which ranges from injunctive to advice-like forms. Procedures of various kinds (social recommendations, do-it-yourself (DIY), maintenance procedures, health care advice, didactic texts) form an excellent source of corpus to observe how explanation are constructed, linguistically realized and what aims they target.

Procedures are of much interest to build a corpus for explanation analysis since the language which is used is often simple and direct. Indeed, procedures are essentially oriented towards action: there must be little space for inferences. This kind of corpus is particularly well-adapted for our investigation and for the development of stable generalizations: it covers quite a large proportion of situations which are reproducible, e.g. in authoring tools, helpdesks or in natural language generation systems.

Explanation occurs also in a large variety of goal-driven but non-procedural contexts, for example, as a means to justify a decision in legal reasoning, in political discourse and debates, as a way to explain the reasons of an accident in insurance accident reports or as a form of synthesis of an evaluation in opinion expression. Explanation may also be associated with various pragmatic effects (irony, emphasis, dramatization,