Human-Machine Dialogue

Design and Challenges

Frédéric Landragin

This book summarizes the main problems posed by the design of human-machine dialogue systems and offers ideas on how to continue along the path towards efficient, realistic and fluid communication between humans and machines.

A culmination of ten years of research, it is based on the author's development, investigation and experimentation covering a multitude of fields, including artificial intelligence, automated language processing, human-machine interfaces and notably multimodal or multimedia interfaces.

Frédéric Landragin is a computer science engineer and has a PhD from the University of Lorraine, France. He is currently in charge of linguistics research for the French National Center for Scientific Research (CNRS). His studies focus on the analysis and modeling of language interpretation. Human-machine dialogue is one of the applications of this research.

Table of Contents

Foreword

The preparation of this book was carried out while preparing an accreditation to supervise research. This is a synthesis covering the past ten years of research, since my doctorate (Landragin, 2004), in the field of human-machine dialogue. The goal here is to outline the theories, methods, techniques and challenges involved in the design of computer programs that are able to understand and produce speech. This synthesis covers the presentation of important works in the field as well as a more personal approach, visible through the choice of the themes explored, for example. How can a machine talk, understand what is said and carry out a conversation close to natural conversation between two human beings? What are the design stages of a human-machine dialogue system? What are the understanding, thinking, and interaction abilities expected from such systems? How should they be implemented? How can we get closer to the realistic and fluid aspect of human dialogue? Can a dialogue system lie?

These questions are at the origin of my path, which oscillated between linguistics and computer science, between pure research and development, between public and private research laboratories: INRIA, then THALES and currently the CNRS. These are also questions that second-year Master students asked me during the human-machine dialogue class that I held at University Paris Diderot for a few years. Thus this book draws inspiration in part from the class preparation and aims to be accessible to a public with linguistic and automatic language processing notions, and not necessarily with knowledge of the human-machine dialogue domain.

The goal here is to explain the main issues created by each stage of the design of a human-machine dialogue system, and to show a few theoretical and technical paths used to deal with these issues. The presentation will not cover all the wealth of existing works, but beyond that, it will aim to provide the readers a glimpse of the field, which might make them want to know more.

The goal here is to show that today there still is a French school of human-machine dialogue, which has been especially active in the past few years, even if it was at times a bit slower and at times it appeared that the human-machine dialogue was an aporia. The French school is characterized by its multidisciplinary approach, its involvement in different fields, such as system development (university prototypes, general public systems, as well as – and we tend to forget them since they are confidential – military systems), implementation of assessment methods and campaigns, and software architecture design. There is a French school for multimodal dialogue, for ergonomics, for embodied conversational agents, and even for the application of machine learning techniques to the human-machine dialogue. Not all the links between these specialties are completely finalized, but the general dynamics are undeniable and encouraging.

As usual in research work, what is presented in this book is indebted to the encouragement, advice and more generally speaking the sharing of an efficient and enjoyable work environment. For their institutional as well as scientific and human encouragement, I would like to thank Francis Corblin, Catherine Fuchs, Valérie Issarny, Jean-Marie Pierrel, Laurent Romary, Jean-Paul Sansonnet, Catherine Schnedecker, Jacques Siroux, Mariët Theune, Bernard Victorri and Anne Vilnat. For the incredibly enriching OZONE experiment during my postdoctorate fellowship at INRIA, I would particularly like to thank Christophe Cérisara, Yves Laprie and especially Alexandre Denis on whom I was able to rely to implement a memorable demonstrator. For the equally memorable experiment of THALES R & T, I would like to thank, more specifically, Claire Fraboulet-Laudy, Bénédicte Goujon, Olivier Grisvard, Jérôme Lard and Célestin Sedogbo. For the wonderful workplace that is the LATTICE laboratory, a Joint Research Unit of the CNRS, I would like to thank, without repeating those whom I have already mentioned, Michel Charolles for our very enriching exchanges on reference, Shirley Carter-Thomas and Sophie Prévost for the information structure, Thierry Poibeau and Isabelle Tellier for natural language processing, my successive colleagues Sylvain, Laure, Frédérique, as well as Benjamin, Denis, Fabien, Jeanne, Julie, Marie-Josèphe, Noalig, Paola, Paul, Pierre and Sylvie. I would also like to thank those with whom I was able to interact through ATALA (I am more specifically thinking of Frédérique, Jean-Luc and Patrick) and within my human-machine dialogue classes, as well as those with whom I have started collaboration, even if they sometimes did not come to fruition. Many thanks then go to Ali, Anne, Gaëlle, Jean-Marie, Joëlle, Meriam, Nathalie and Tien. Finally, I would like to thank to Céline for her constant encouragement and unending support.

Frédéric Landragin

Introduction

The OZONE (Issarny *et al.*, 2005) system mentioned in the Foreword was a demonstrator for a train ticket reservation service within the framework of the European OZONE project. It is a recurring *application* (or *task*) in human-machine dialogue, and this is the framework that we will use to provide examples throughout the book. The computer program behind the demonstrator was able to process an audio input, transcribe the captured speech into text and understand the text in order to provide an adequate answer. The task required the system to know the timetables of a set of trains in a given region, and so a database was implemented: it allowed the dialogue system to find crucial information for its answers, which, as in a human dialogue, were given orally. Until now, we have remained within the framework of the *human-machine spoken dialogue*, which has vocal inputs and outputs. This type of system can be used on the phone with no visual channel. Ideally, the system is quick, comprehensive and provides relevant answers, so that the user has the impression he is talking spontaneously, as with a human interlocutor.

However, we had set ourselves an additional specification, that of creating a *multimodal* system able to manage both speech and pointing gestures carried out on a touch screen. The system was able to recognize pointing gestures and to link these gestures with words pronounced simultaneously. What was true of the system's input had to be true of its output as well, and thus we have designed a system able to manage output multimodality, which meant it could produce both a vocal utterance and a display on the screen. In other words, once the system had decided on an answer to give the user, it could decide either to verbalize its answer, to display it on the screen, or better yet verbalize part of it and display the rest. This is what we call a *multimedia information presentation*. Going beyond the issues with oral dialogue, we have reached the issues in multimodal dialogue. The systems in question involve a *communication situation* shared between the human user and the machine. This shared situation brings together a visual context (what appears on the computer's screen) and gestures (which remain very simple for now since they are limited to the contact on the screen). With this

communication situation, we get closer to in-person human dialogue: the user faces the machine when speaking and sees a visual display that the machine also "sees".

To work, the program thus had to be carried out on a computer with at least a microphone, a speaker and a touch screen, which was much less common in 2004 than it is now. Figure 1 shows an example of the dialogue that the system could have with a user. The successive speaking turns are shown with a letter (U for user, S for system) and a number to identify the analyses and discussions more easily.

	Utterance	Action on the screen
S1:	"*Hello, I am the train ticket reservation system.*"	Display of a map on screen
U1:	"*Hello, I would like to go to Paris.*"	–
S2:	"*Here are your possible itineraries.*"	Two itineraries appear
U2:	"*How long with this itinerary which seems shorter?*"	Gesture pointing to one of the itineraries
S3:	"*Twenty minutes.*"	Highlighting the chosen itinerary
U4:	"*Very well, I would like to book a single journey.*"	–
S4:	...	...

Figure 1. *Human-machine dialogue example*

A dialogue like this one is a type of *discourse* – that is a series of sentences linked each other – with the specificity that it only involves two speakers and not only one. When a dialogue involves more than two speakers, we can refer to it as a *multilogue*. If we take the succession of words "here are your possible itineraries", we use the term *sentence* as long as we take these words, their organization and their meaning out of context, and the term *utterance* if we take the context into account, that is the fact that this sentence was uttered by the system S at a specific moment in the dialogue, and, in this case, at the same time as a display action (which gives the word "here" a specific meaning, and this word is meant to present multimedia information). Depending on the context, a sentence can be the source of various utterances.

The example in figure 1 is an *interaction*, according to the terminology adopted. In S1, the system presents itself; then, from U1 to U4, the dialogue focuses on the purchase of a train ticket. The extract from U1 to U4 is an *exchange*: the goal defined in U1 is reached in U4, which closes the exchange without putting an end to the interaction. An exchange necessarily involves both speakers, and has various speaking turns, at least two. S1, U1 ... U4 are *interventions* that match the speaking turns. An intervention only involves a single speaker and defines itself as the biggest monologal unit in an exchange. An intervention can be understood as a single *speech act* (action performed by speech, such as giving an order and answering a question), such as in

S2 or S3, or various speech acts, such as in S1 or U1 where the first act is a greeting and the second act is the transmission of information.

Based on the use of language (or natural language opposed to the artificial languages of computer science), the dialogue is studied due to notions of linguistics. The analysis of utterances thus falls within the field of pragmatics, a study of the language in use. The analysis of the sentences themselves falls within the field of linguistics. More specifically, the analysis of the meaning of sentences and concepts involved falls within the field of semantics. At the level of sentence construction, we focus on words, on units that create the lexicon, on groups of words, on the order in which they appear and the relations between them, which is syntax. In an oral dialogue, we also focus on the phonic materialization of sentences, the prominences, the rhythm and the melody, which falls within the field of prosody. To all these analytical outlines, we can add all the phenomena characterizing natural language, especially the fact that there are a variety of ways to express a single meaning, or that the language is in its essence vague and imprecise, which can lead to *ambiguities* (more than one interpretation of an utterance is possible) and *underspecification* (the interpretation of an utterance can be incomplete). This is the wealth and diversity of language, which a *natural language dialogue* system needs to take into account if it wants to be comprehensive. Language in a dialogue situation is characterized by wealth and diversity which are notably expressed through utterance combinations, i.e. the way in which an utterance is linked to the previous one, and the way in which various successive utterances create an exchange, and, in a general manner, in the dialogue structure which, builds itself along with the interaction, and is also an object of analysis. When this structure does not reflect a rigidity of codified protocol but a natural use of the language, we reach a final definition, that of *natural dialogue in a natural language*.

This is the field of research and development covered in this book, and it has been already explored in many books, whether as is the aspect of system presentations or of sufficiently formal theories which in the end authorize computer implementation. As an example, and in chronological order, we will mention a set of books whose reading is useful, even crucial, for any specialist in the field of human-machine dialogue: Reichman (1985), Pierrel (1987), Sabah (1989), Carberry (1990), Bilange (1992), Kolski (1993), Luzzati (1995), Bernsen *et al.* (1998), Reiter and Dale (2000), Asher and Lascarides (2003), Cohen *et al.* (2004), Harris (2004), McTear (2004), López-Cózar Delgado and Araki (2005), Caelen and Xuereb (2007), Jurafsky and Martin (2009), Jokinen and McTear (2010), Rieser and Lemon (2011), Ginzburg (2012) and Kühnel (2012). To provide the reader with a few points of reference and approach the main aspects of the field, we will give a chronological outline of the field's history in Chapter 1.

The field of human-machine dialogue covers various scientific disciplines. We have mentioned computer and language sciences, but we will also see in Chapter 2 that other disciplines can provide theories and supplementary points of view. With the

aim of designing a machine that has abilities close to a human being (we try to get as close to human abilities as possible, without simulating them), we can find inspiration in all kinds of studies focusing on language and dialogue so as to model them in a computational framework which would allow for their use in human-machine dialogue.

The field of human-machine dialogue (from now on HMD) has links with other fields, such as natural language processing (NLP), of which it is an essential application; artificial intelligence (AI), from which it arises and which completes the linguistic aspects with the reasoning and decision-making aspects; human-machine interfaces (HMIs), which it helps enrich by offering vocal interaction possibilities in addition to graphical and touch screen interactions; and, more recently, question-answering systems (QAS) and embodied conversational agents (ECAs), which are some of its aspects – the first to focus on the natural language interrogation of large databases and the second on the visual and vocal rendering of the avatar representing the machine-interlocutor – which have become fully fledged research fields. The HMD field thus brings together various issues that can be separated into three major categories:

– processing signals at the system's input, with automatic recognition and interpretation;

– the system's internal processes and reasoning;

– managing the messages generated by the system, i.e. at its output, with automatic generation and multimedia information presentation.

According to the type of system considered (tool versus partner, or to put it differently, by offering the user a logic of *doing* or of *making do*), according to the communication modalities between user and system (written dialogue versus oral), according to the part given to a task underpinning the dialogue (dialogue in an open domain versus closed domain) according to the importance given to the language (dialogue favoring the task versus dialogue favoring linguistic fluidity and realism), these issues give rise to many approaches and ways of implementation. The approaches can be rather theoretical – for example extending and testing a syntactic or particular pragmatic theory – or rather practical (favoring robustness). The implementations can be rather symbolical or rather statistical, etc. Chapter 3 will review these aspects describing the stages of achievements of the HMD system. As for the question of software architecture, chapter 4 will complete and finish the first part of the book with the crucial challenges such as reusability and design of generic models, like what is being done in the field of HMI.

Processing utterances at the system's input is the focus of our second part, with chapter 5 looking at the fundamental lexical, syntactic, prosodic and semantic aspects, chapter 6 at the issue of resolving contextual reference and chapter 7 the recognition and interpretation of speech acts in the context of a dialogue. We will quickly go over the questions of automatic recognition of speech and the so-called *low level* processes

to focus on the *high level* processes which revolve around the meaning of utterances: semantics, reference, speech acts. With the example of U2 in Figure 1, the chapter which focuses on semantic analysis will show how representing the significance of the sentence "how long with this itinerary which seems shorter?", a complex sentence since it has a main clause and a subordinate clause, and the main clause has no verb. Without such a linguistic analysis, a HMD system can hardly be called comprehensive. The chapter focused on the reference will show how the utterance and the pointing gesture of U2 allow us to provide the demonstrative referring expression "this itinerary" with a *referent*, in this case a specific train journey. Without this ability to solve reference, a HMD system can hardly know what is referred to in the dialogue. The chapter focusing on the speech acts will show how this U2 intervention can be interpreted as a set of two speech acts, the first act being a question, and the second act commenting the referred train journey, a comment which can then be processed in different ways by the system, for example if it is indeed, or not, the shortest itinerary. In this case again, the chapter will highlight an essential aspect of an HMD system: without an ability to identify speech acts, a system can hardly know how to react and answer the user.

The system's internal and output processing determines its behavior and are the focus of the third part of this book. In chapter 8, we will see how identifying speech acts allows the system to reason according to the acts identified, to the task and the dialogue already carried out. This question highlights the issue of putting the user's utterance into perspective and determining the appropriate reaction in return. Beyond all the processes studied in the second part, this is where we have to reason not at the level of a single utterance but at that of the dialogue as a whole. We will thus speak of *dialogue management*. In chapter 9, we will see how a system can carry out the reaction it has decided on. It is the question of automatic message generation, a question which takes a specific direction when we take into account avatars (and we join here the field of ECA), or even, much more simply, as we have mentioned it before, the possibility of presenting information on a screen at the same time as a message is verbalized.

Finally, chapter 10 will deal with an aspect that concerns the design stages as well as the final system once the computer implementation is finished. It is the question of evaluation, a delicate question inasmuch as a HMD system integrates components with various functionalities, and in which, as we have seen, the system types that can be considered have themselves highly varied priorities and characteristics. This question will lead us to conclude on the field of HMD, on the current state of achievement and the challenges for the years to come.

FIRST PART

Historical and Methodological Landmarks

Chapter 1

An assessment of the evolution of research and systems

Human-machine dialogue (HMD) systems appear more present in the works of science-fiction than in reality. How many movies do we know which show computers, robots, or even fridges and toys for children who can talk and understand what they are told? Reality is more complex: some products that have come from new technologies, such as cell phones or robot companions, talk and understand a few words, but they are far from the natural dialogue which science-fiction has been promising for years.

The ideas for application are not lacking. Implementing a dialogue with a machine could be useful for getting targeted information, and this could be for any type of information: transportation (Lamel *et al.*, 2000), various stores, tourist or recreational activities (Singh *et al.*, 2002), library collections, financial administrative procedures (Cohen *et al.*, 2004), etc., see (Gardent and Pierrel, 2002) and (Grau and Magnini, 2005). The dialogue is indeed adapted to the step-by-step elaboration of a request, a request that would be difficult to hold in a single utterance, or in a command expressed in a computer language. The first field of application of HMD which includes QAS is sometimes defined as *information-seeking dialogue*. When the dialogue only concerns a single topic, for example railway information, we talk of closed-domain dialogue. When the dialogue can be about pretty much anything, for example the questioning of the encyclopedic database as IBM Watson recently did with a TV show task, we talk of open-domain dialogue (Rosset, 2008). If we reuse the example of the introduction, a unique utterance with no dialogue could be as follows: "I would like to book a single journey to Paris taking the shortest itinerary as long as it takes less than half an hour (otherwise I do not wish to make a reservation)". The elaboration of a natural dialogue is much more flexible: it allows the user to express a first simple request and then improve it according to the machine's answer; it allows the machine

to transfer information for a future action, and confirm or negate along the way (Pierrel, 1987). The total number of words to arrive at the same result might be greater, but the spontaneity of the utterances and their speed as well as the ease of production is more than fair compensation. With the example of questioning a yellow-page-style directory, (Luzzati, 1995) shows another advantage of dialogue: the user can obtain the address of a taxidermist even when he does not know the name of this profession. Through the conversation, the dialogue, the user gets the machine to understand exactly what he is looking for. There is a joint construction of a common concept to both interlocutors, and this joint construction is the point of the dialogue compared to the unique utterance or the computer language request.

Beyond the information request or the consultation of a database, installing a dialogue with a machine can also be useful to manage a computer system, for example digital design software (drawing, image processing, 3D) or simply a computer's operating system. We can also imagine that instead of looking for the accurate function in the numerous menus and submenus of the software in question, the user carries out vocal commands that are much swifter and more direct, at least if he is not familiar with the software. This second field of application of HMD is close to that of HMI, and is sometimes defined as *control command dialogue*. Including the computer science software development, we can almost imagine a user who would use the language to *program in natural language* (Luzzati, 1995). Including robotics, this is the field of robot command, the key application of modern AI (Gorostiza and Salichs, 2011). Moreover, it is also the field of professional, civil and military systems whose design I took part in at THALES: air traffic control and management, maritime surveillance, supervision of the situation on the field and system command in dangerous areas. These systems are currently complex HMI and the research team's work in which I participated was to test the potential of *giving them speech.* We remained there in the closed-domain control command dialogue, but with many robustness limits.

Information dialogue, control command dialogue: all the existing systems will not fall into one or the other of these strict categories. Some systems allow for both types of interaction, for example some companion robots can both give their users information and carry out simple tasks on demand, such as walking or dancing. Other systems do not aim to provide information or carry out specific tasks. These are, for example, purely recreational systems, with the example of conversation robots on the Internet. The examples given here are taken from general public HMD systems, or at least systems that are destined to be such. But do they really work? In fact, what we can note when using this type of system is that it is quite difficult to establish a proper dialogue. When a word is recognized and understood, which is not systematic, the machine tries to give an answer based on this word or attempts to restart the dialogue in its own way, which is rarely a relevant one. As (Vilnat, 2005, p. 5) states, the HMD systems only work in a very imperfect manner and thus are greatly criticized, up to the point where "it will never work" is often heard. The criticism comes, first and foremost, from the users who notice that there is a wide gap between what they test

and what they hope for, and they often believe that a classic HMI is quicker, more efficient and even easier, or less confusing, to use. The criticism also comes from researchers and developers in the HMD field. Indeed, the amount of work required to achieve a system is such that there is a lot of discouragement. The amount of work corresponding to a doctoral dissertation is not sufficient, at least when trying to achieve an innovative system. As an example, (Guibert, 2010, p. 60)'s discouragement when designing a system called *A* is striking: "following the termination of the development of this system A, taken as an example among others, this body of work is actually the chronicles of the foretold failure of current dialogue systems."

We will see that when the dialogue is directed by a clearly defined task, it is possible to design a performing HMD and this design has actually greatly progressed in the past few years. After discussing a few historical landmarks (section 1.1), we will quickly cover the functionalities that are more and more present on current systems (section 1.2), and from this we will deduce a primary list of potential challenges for the years to come (section 1.3).

1.1. A few essential historical landmarks

The dialogue between human being and machine is a key field in computer science: a kind of quest for the Holy Grail, which was the source of computer science developments and researcher vocations. As it so happens, the first system to become a landmark, ELIZA (Weizenbaum, 1966), is also a huge subterfuge (which was assumed, as we will see page 26). Various paths were then taken in serious HMD system design: a path close to AI, with a focus on interpretation and reasoning issues, and a path that consisted of enriching the automatic speech recognition systems. Both paths with their two separate communities (Vilnat, 2005, p. 47), have recently come together again and allowed various consistent HMD systems to reach fruition. These are the systems we will now present.

1.1.1. *First motivations, first written systems*

Can a machine think? In 1950, Alan Turing relaunched this question which had recurred throughout technology's history: he substituted the question "can a machine imitate a human?" and suggested a game, or test, based on imitation, which became famous by the name of the Turing test. At first, the imitation concerns a man and a woman: the test subject talks with a man and a woman in turn, through machine-typed pieces of paper, without seeing or knowing anything about his successive interlocutors. The man has to try and pass for a woman, and the subject thus has to guess which one is the man and which one is the woman. Then, without the subject's knowledge, the man is replaced by a machine. If the subject cannot identify either of the interlocutors, then the machine passed the Turing test. This game, created at a time when it was

impossible to program an HMD system, was the source of innumerable discussions, of various and varied assertions on the nature of the machine or of the human being. The interesting thing here is the challenge for computer science: to program an HMD system that can be thought to be a human being. A. Turing does not give us many hints as to how to achieve that result. The description of the test is focused on experimental conditions and does not address the importance of language and dialogue in this approach to thought (Tellier and Steedman, 2009). Nonetheless, there are competitions organized today (such as the Loebner prize) inspired by the Turing test. The 1950s correspond to the first research motivations for HMD, information seeking and NLP. We should point out that the ATALA association, created in 1959 was originally called *Association for the study and development of automatic translation and applied linguistics* ("Association pour l'étude et le développement de la Traduction Automatique et de la Linguistique Appliquée" in French) and then became the *Association for Natural Language Processing* ("Association pour le Traitement Automatique des Langues").

The 1960s mark the appearance of the first HMD systems. ELIZA[1] (Weizenbaum, 1966), which we mentioned earlier, is fascinating in more than one way. First of all, this is a written dialogue system that really works without looping or randomly stopping. It is always possible to carry out conversations on hundreds of speaking turns. Moreover, the chosen task is itself fascinating: the system is supposed to play the role of a non-directing psychotherapist, which means it simply listens to the speaker to tell it about his problems ("I have a problem with my parents") and sometimes reacts to certain sentence ("tell me about your family"). The realism is so strong that some users have spent hours talking with ELIZA, and J. Weizenbaum had to decide against openly adding a dialogue-saving module, faced with accusations of spying and violating privacy. This task has two advantages: it does not have to carry out a complex dialogue, for example with negotiation or argumentation, while keeping a spontaneous and natural aspect, since the user can say what he wants when he wants to; and on the other hand, it is easy to program, since the system does not need to understand absolutely everything: utterances such as "what makes you say that?" or "I see, please go on" are vastly sufficient. Indeed, and this is fascinating for NLP, AI or HMD researchers, J. Weizenbaum managed to develop a system that appears to master language and pass the Turing test, whereas it does not even approach the most basic issues of automatic understanding.

1. The name came from the Eliza Doolittle character in the movie *My Fair Lady* (1964, G. Cukor), itself an adaptation of the play *Pygmalion* (1914, G.B. Shaw), which has also been adapted for the movies. Eliza Doolittle is a florist from a very poor neighborhood, and becomes the subject of a bet when an aristocrat claims that by changing her manner of speech, he will be able to make her pass for an aristocrat herself. The idea of duping someone through language and dialogue is thus the origin of the system's name.

Indeed, all of ELIZA's operating relies on a few well-chosen heuristic rules. The system knows a few words, especially those linked to family: "parents", "mother" and "father". It is thus able to bounce off the utterance "I have a problem with my parents" without any understanding involved in this process: the system just detected "parents" and answered with a new question on "family", a new question that actually allows it not to have to take into account the meaning of the user's utterance. The system also knows the personal pronouns referring to the two interlocutors, "I", "me", "my", "you" and "your" that allows it to carry out replacements and build an utterance taking up parts of the user's utterance, such as "what makes you believe that *you* are listening to *my* advice?", generated after "*I* am listening to *your* advice". With this example, we can note that the system does not understand much, but it is able to switch the persons around and frame the user's utterance in a question "what makes you believe that?", a deliberately open question. The techniques implemented by the input utterances are word sequence detection and keyword detection. Those implemented for the system's output utterance generation are the direct production of typical sentences, the concatenation of text span, whether they are typical spans or spans obtained through a user's utterance. The system also has the beginning of memory, inasmuch as it is able to return to a familiar term used a few speaking turns prior.

A few years after ELIZA, the PARRY (Colby *et al.*, 1971) system had an impact due to its supplementary techniques. This time the machine simulates a paranoid subject during his first (written) interview with the user who is supposed to play the role of a psychiatrist, a profession to which the main author incidentally belongs. The claimed scientific approach is the studying and modeling of paranoia, and this goes so far as the funding that comes in part from the National Institute of Mental Health, and the methodology that includes not only the modeling and computer science development of the model, but also its assessment by mental health professionals: a total of 25 psychiatrists were involved, and the overwhelming majority of them (23) diagnosed the system as paranoid, making it pass the Turing test with flying colors. The dialogues are carried out as interviews and start with the factual questions that the user asked the system: name, age, occupation. Thus, PARRY has in his memory a set of answers to these typical questions: his name is Frank Smith, he is 28 years old, and interned in a hospital. He also has in his memory various questions that the system can ask, thus inverting the dialogue orientation: "who are you?", "what do you want with me?", as well as anecdotes, and especially words around a relatively well-elaborated concept, such as that of *mafia.* The techniques implemented are also techniques of text span research, keyword detection, first and second person pronoun management, but all with more finesse than ELIZA had. For example, the word "fear" has a set of predefined spans, and verbs such as "to believe" have specific processes. Moreover, the system is characterized by an attempt at personality or *mental states* through variables: fear, anger and distrust. The values of these variables increase or decrease as the dialogue unfolds, according to what the user says. The system's behavior evolves in a consequent manner: it becomes aggressive if the anger value passes a certain threshold. The

rules or heuristics, on the contrary from ELIZA's rules, are based both on the user's utterances and on the variables of state. PARRY marks an evolution of HMD systems, with the technical means of the time: the program, written in a variant of the LISP language, takes 35 KB of which 14 KB belong to the database.

The 1970s were the time of the first (written) understanding systems, with significant improvements in NLP, especially in syntactic and semantic analyses and, thus, the first true systems of written HMD that model a field of knowledge, know how to interpret an utterance in this field, and start to manage a structure dialogue. This progress follows a few landmark works in linguistics and computational linguistics, especially B.J. Grosz and then C.L. Sidner, as (Jurafsky and Martin, 2009, p. 892) underline it. That corresponds to the first path mentioned on page 25, with two key systems, SHRDLU[2] and Genial Understanding System (GUS). In parallel, the speech recognition system path is also progressing strongly, especially with systems developed within the American Advanced Research Projects Agency (ARPA) projects: HARPY, HEARSAY, HWIM. We thus go from the recognition of isolated words, which is not at all adapted to HMD, to the recognition of continuous, and eventually multi-locutor words, with concerns which start to reach those of HMD, for example the question of software architecture to get various sources of knowledge communicating inside systems, see (Pierrel, 1987)'s historical outline. We will return to this in paragraph 1.1.2 with the first oral HMD systems.

The SHRDLU system (Winograd, 1972) gives a new boost to written HMD by showing the deeper understanding and dialogue possibilities as soon as you limit yourself to a clearly limited and modeled task. This time, let us forget about the Turing test and turn to targeted applications: the task consists in displacing geometrical objects (cubes, cones and pyramids) with a machine. It involves the display of a scene on a screen, with a representation of the system itself with a kind of robot arm manipulating objects. The user creates utterances such as "pick up a green block" or "find a block which is taller than the one you are holding and put it into the box", and the system carries out these actions, whose results are visible on screen. This task puts the accent on *object reference* phenomena: what object is referred to by "the pyramid"? To correctly interpret such a reference, the system must find among the objects on display which one is the correct one, meaning which one corresponds to the user's intent. If two or three pyramids are visible, the system can thus answer "I do not understand which pyramid you mean". After clarification, it does what it must do, that is carry out the actions and answer questions. Many of the possible questions revolve around the physical world of objects: "what does the box contain?", "what is the pyramid supported by?". Each time, SHRDLU is able to analyze the scene, identify the spatial

2. The name comes from the sequence of letters E T A O I N S H R D L U that is, in decreasing order, the sequence of letters most often used in English, in the way they are vertically shown in the middle of some printing machine keyboards.

relations between objects, count and answer. Certainly, a world of geometric objects remains simple. But all these implemented automatic understanding processes are impressive, as well as the matching knowledge modeling: the system is able to solve complex references, such as "a block which is taller than the one you are holding", to solve anaphora such as "put *it* in the box", to identify speech acts. The resulting dialogue is focused on the essential. There may be a lack of fluidity, but the goal is to satisfy the task, and indeed, all is done for this to happen.

As for GUS (Bobrow *et al.*, 1977), it takes an additional step into the utilitarian HMD, with a flight reservation task. To demonstrate this research prototype, the database only comprises a single flight in California. Beyond this limitation, the linguistic modeling, the computer modeling and the methodological aspects give an idea of what the HMD domain will look like in a few years. Just as SHRDLU, the system is able to solve object and anaphora references, at least when they directly concern the task's objects, that is the flights, days and timetables. For example, it manages to allocate a date-type reference to the referring expression "Friday" used as a return date after specifying "May 28th" as an outward flight. The interpretation of the user's utterances triggers a syntactic and semantic analysis that can be partial, and thus operate on other linguistic materials than just full sentences. It also triggers a recognition of speech acts, notably with the understanding of indirect answers to some questions. The great results of linguistic works on the dialogue structure and the information structure are used, which leads to the system managing a great deal of knowledge on language: lexicon (3000 roots recorded, which is greater than the precedent systems), morphological rules, syntactic constructions, simplified principles of the information structure, patterns for the dialogue structure, conceptual models for the travel plans and date, and finally the agenda model: central structure that will allow the system to manage events and know at any moment what task to carry out. The computer implementation is being rationalized: the different linguistic analyzers are implemented as independent modules and a communication language between the modules is specified. The fact that the modules are independent allows us to test them, correct them and improve them separately. Actually, all design follows an exemplary methodology: the authors have started by collecting and studying the human dialogues focusing on the same task, that is they carried out a corpus study, the word *corpus* referring to a collection of attested linguistic material, and they even implemented a system simulation experiment (which will later be called a *Wizard of Oz*), to collect the data on the user's behavior when faced with the system they imagined. The fundamental methods of the HMD are set. Obviously, they are applied with the means at the times, and the computer's sluggishness, for example, leads to a wait between 10 and 60 seconds for each utterance, a wait which is taken into account in the simulation experiment, and is very far from the speed and naturalness of human dialogue.

1.1.2. *First oral and multimodal systems*

While it was possible up until now to present the major advances in HMD through a few emblematic systems, this has no longer been the case after the 1980s. Indeed, this decade saw a multitude of theoretical works that many researchers discovered, the dialogue and its characteristics, a multitude of prototypes and HMD systems, and notably the first oral systems and the first multimodal systems. Moreover, it was also the golden age of video games and the general public discovered adventure games with textual interaction[3], that were the first recreational HMD systems.

Among the theoretical works that have marked the 1980s, there is the research carried out by conversational analysis (Sacks *et al.*, 1974) and discourse analysis or discourse pragmatics (Roulet *et al.*, 1985; Moeschler, 1985). Although the objectives of these two studies differ, their focus – i.e. the recording and transcription of human dialogues – is the same, and the observations will give us a clearer view of the notions defined in the introduction and how they relate to one another (speaking turn, intervention, utterance and speech act). They will also help in the comprehension of the notions of cooperation, planning, conversational organization, dialogue structure, common ground, grounding and relevance, which we will see in Chapter 8. These works would contribute to numerous articles being published (Allen and Perrault, 1980; Clark and Wilkes-Gibbs, 1986; Grosz and Sidner, 1986; Clark and Schaefer, 1989; Cohen and Levesque, 1990) that would inspire the whole HMD community.

The first oral systems arise from the progress in automatic speech recognition. To operate correctly, they focus on well-defined tasks, like SHRDLU and GUS do. For example, the NUANCE company develops various specialized systems, often for telephone dialogue for clients such as banks (Cohen *et al.*, 2004). As for the first multimodal systems, i.e. systems that match speech recognition with gesture recording which at first corresponded to simple clicks of a mouse, they appear in a famous article (Bolt, 1980), which shows that multimodality is much more efficient than just speech to refer to objects, as long as the HMD system involves a visual scene. A new side of the HMD field was then opened, new questions were asked on ergonomics, on the

3. As an example, the SRAM (*Mars* backwards) game, published in 1986 by ERE Informatique, left a strong impression on the 12-year-old player that I was at the time: all the interaction in the game went through written commands, which led me to type, for example, "I want to go west" and see (visualization of the analysis steps carried out by the software) the utterance appear on screen with the words "go" and "west" highlighted in color, and then discover the software's answer: "you arrive near a waterfall", with the display of a visual scene in which the player must look for clues to continue his quest. The techniques implemented here are much simpler than those in SHRDLU or in GUS, with keyword detection instead of word sequence detection, and the keywords are almost always found within the verb–complement pattern, but the limits are not the same: the vocabulary and possibilities are vast, adapted to intensive use, and the game must also be robust, reliable and interesting.

spontaneity of multimodal dialogue, on interactions between HMI and HMD, and in general on all the inputs and limits of multimodality, see a summary in (Oviatt, 1999). Among these questions, the following opened new perspectives: if an HMD system is able to carry out automatic interpretation taking multimodality into account, should it not carry out automatic generation also taking multimodality into account? With demonstrators, for example in the field of air control, we have begun to explore this issue of output multimodality and outline its own field of research, that of IMMPS (Intelligent MultiMedia Presentation Systems, see Chapter 9).

The 1980s are thus full of questions. After the first systems fascinated and helped clarify the methodology and limits of HMD, they gave way to natural dialogue in natural language with new goals such as spontaneous speech processing, gesture recording and use of interaction devices with all that it implies: contextual management, adaptation to the display device and adaptation to the user.

The 1990s kept on the same path by broadening the panel of expected functionalities in an HMD system. This decade corresponded first to the entrance of the digital age, the consequences of which were primarily a renewal in theoretical and experimental research on spontaneous oral language, and second to the introduction of programming techniques based on important calculations, especially probabilistic calculations, which were costly in computer resources. Research on oral language was until then hindered by technical constraints, but the digital world greatly helped promote the rise of oral analysis software, the multiplication of studies and finally a change in point of view on oral language, which acquired the status of full-fledged study subject and not just a poor child of the written language, or even a poor child full of mistakes (Blanche-Benveniste, 2010). The consequences for HMD are that the work is not only based on grammars and rules stemming from the written language; little by little the specificities of oral speech are integrated: corrections, repetitions, inserted clauses, as we will see in paragraph 5.1.2. On the other hand, the use of speech input creates new issues for HMD systems, with, for example, the need for the user to use a key or pedal at the same time as he speaks (push-to-talk), to let the system know the beginning and end of his utterance. As for programming techniques, they are enriched by advances in statistical approaches that integrate probabilities calculated from a corpus that, as (Jurafsky and Martin, 2009, p. 892) underline, starts the probabilistic processing of speech acts, and brings a supplement to HMD system realizations, which goes beyond the quality of previous research prototypes. Efforts were also made to enrich the automatic understanding methods, with joint approaches that combine both *bottom-up techniques* (starting from the utterance, the system carries out various analyses to identify the underlying meaning and intention) and *top-down techniques* (starting from the possible plans and intentions, the system carries out various analyses to determine which intention satisfies the utterance). These efforts involve research on the representation of plans and reasoning, which presupposes that the system manages to reason on the beliefs of the user (Vilnat, 2005, p. 6). We then see models of the BDI (Belief, Desire, Intention) type appear.

Among the systems in the 1990s, the TRAINS system (Allen *et al.*, 1995) is exemplary since it tries to find solutions to a vast panel of challenges around automatic understanding, of a dialogue with joint initiative (not solely commanded by the system or the user), of representation and reasoning on time, actions and events. The task falls into the domain of transportation, but unlike GUS or the example we gave in the introduction, it involves various modes of transportation and thus manages the connections between these modes, the planning issues, the optimizations (journey length calculations), the potential conflicts, etc.

In France, systems and publications are multiplying (Bilange, 1992; Guyomard *et al.*, 1993; Duermael, 1994; Luzzati, 1995; Sabah *et al.*, 1997; Grisvard, 2000) and we will remember as an example the DIALORS system by D. Luzzati, which focuses once more on train ticket reservation. The methodology starts here again with an in-depth corpus study, in this case a corpus coming from the National Society of French Railways (SNCF, Société Nationale des Chemins de Fer Français) recordings, a corpus that has also been the focus of various publications. The DIALORS system has an analyzer called ALORS whose function is to turn utterances into an internal representation to the system, and a dialogue manager, DIALOG, who decides, depending on the representation, on the action to be carried out: request clarification, answer the user's query after consulting the train timetable database. The second component has the role of implementing the dialogue model suggested by the author, a model that distinguishes the *governing dialogue*, i.e. the main dialogue reflecting the task's progression, from other potential *incidental dialogues*, i.e. the clarification requests and other transient sub-dialogues, which do not influence the task's progression but allow the interlocutors to understand each other. This dialogue structure allows the system to carry out fine analysis and also to assess in real time the task's progression, without requiring the implementation of a more complex model such as the hierarchical model of the school of Geneva (Roulet *et al.*, 1985), mentioned earlier as an approach to discourse analysis.

1.1.3. *Current systems: multiplicity of fields and techniques*

Our overview started in the 1950s and now reaches the 2000s. It is harder to use hindsight on this period that includes the current systems, especially since the work has multiplied and the number of techniques has increased. In general, beyond the improvement of all the models of the 1990s (Jurafsky and Martin, 2009, p. 892), here is what appeared in the 2000s:

– the application of computer techniques of machine learning to HMD to relocate part of the different settings onto the big corpus processing or onto an improvement of the performances as the system is used (Rieser and Lemon, 2011);

– the tremendous efforts of standardization: W3C, ISO, TEI, DAMSL, etc.;

– the increase in system assessment methodology (see Chapter 10);

– the multiplication of communication modalities with the machine, and thus of the models and techniques of multimodal dialogue: force-feedback gesture or haptic gesture, gestures and postures caught on camera, taking into account the eye direction, lip reading, etc. (López-Cózar Delgado and Araki, 2005);

– the implementation of links with other scientific domains, such as robotics (see Chapter 10 of Garbay and Kayser, 2011), and other fields of NLP, for example machine translation within HMD systems, being able to go from one language to another (López-Cózar Delgado and Araki, 2005);

– the increase in *toolkits* for a quick prototyping of HMD systems, for example the well-known VoiceXML, a standardized language for relatively simple, from a linguistic point of view, voice applications;

– the integration of HMD in wide intercommunication platforms, whether we are referring to ambient intelligence or other aspects, for example linked to software architectures (Issarny *et al.*, 2005);

– the rise of the ECA field that takes into account the emotions;

– the rise in the QAS field.

About this last item, for example, the point of integrating dialogue abilities to a QAS is to allow it to carry out exchanges to specify bit by bit the query (Vilnat, 2005, p. 48). From a (single) question-answer system that is content with finding the result to a query, like the database managers do or like IBM Watson that is still limited by the rules of a game show, we move on to a system of questions and answers (plural), in which the dialogue allows for clarifications, precision, and especially follow-up questions on the same subject: “does this journey go through Meudon?”, “is this the shortest journey to get to Paris?”, “when does it leave?”.

An example for such a system using HMD and QAS is the RITEL project (van Schooten *et al.*, 2007; Rosset, 2008). The system’s architecture highlights question management, with modules devoted to topic detection, user return management, dialogue history management, question routing, implicit confirmation management and additional query management. The project’s goals clearly highlight the QAS performances as much as the HMD performances, and the project is therefore a significant step for open-domain HMD systems that started to emerge in the 2000s. As an example, to compare with the figures mentioned previously in this chapter, RITEL’s vocabulary has 65,000 words, which approximately matches the number of entries in a language dictionary. Another example from the 2000s, the AMITIÉS system (Hardy *et al.*, 2006), a closed-domain HMD system provides us with a way to compare previous systems of the same type as well as an open-domain system such as RITEL. AMITIÉS was designed from an in-depth corpus study of about 1,000 dialogues all belonging to the financial domain on which one of the tasks is focused. The figures corresponding

to this material are as follows: 30,000 sentences for approximately 8,000 words of vocabulary. This is much more than GUS could do, but is still very far from the 65,000 words of a language.

Finally, in the 2000s (and in the following decade), as we saw at the very beginning of this chapter, the first general public HMD systems have appeared, incorporated to various Websites, electronic diaries, geolocation systems and other personal digital assistants. Even if the quality is not there yet, we can imagine that it will help encourage the scientific community's efforts.

1.2. A list of possible abilities for a current system

At the level of general public systems, as we mentioned, we are still far from a natural dialogue in natural language. A few tests of systems, called *voice-controlled* or *voice-recognition* systems, allow us to quickly verify this. For example, the geolocation systems and cell phones are still at a keyword detection level: city names for the first and recipient names for the second. We are still very far from the automatic understanding of utterances such as "I want to go to Grenoble by bypassing Lyon and avoiding the highway between Saint-Etienne and Lyon", in which the user mentions a point of passage and different preferences for two parts of the journey all at once (a much quicker request to say than to program directly into the system, if at all possible). We must still admit that from examples such as these, voice control is not often adapted to the computer system user: it is often noisy, we are never sure of being understood properly, and we are always convinced of being more efficient by directly manipulating the system with a classic HMI. Contrary to what various researchers claimed in the 1980s, one cannot say that because there are more and more computers and more and more data accessible that the HMD will impose itself as a new communication mode. As (Vilnat, 2005, p. 5) states, the question should instead be to know for which tasks it would be useful to implement a dialogue rather than any other kind of interaction technique: the major hindrance is the low interest of users in using an HMD system.

At the level of research prototypes, the natural dialogue in natural language becomes feasible, at least within the framework of a targeted task. This is the case for the closed-domain dialogue and also for some open-domain demonstrators such as IBM Watson. One should however note that the recent endeavors have focused on broadening the systems' abilities rather than developing NLP aspects. We will see this in the three parts matching the three characteristics of a cognitive system: input processing (paragraph 1.2.1), the system's internal analyses (paragraph 1.2.2) and output management (paragraph 1.2.3).

1.2.1. *Recording devices and their use*

Chapter 2 of (López-Cózar Delgado and Araki, 2005) draws an exhaustive list of the multimodal HMD systems with processes carried out on inputs. Without drawing up such a list again, let us quickly mention the following recordings: speech recording; lip-reading the user to help or even replace speech recognition (noisy environment, disabled user, whispering); user recognition; face location and tracking, as well as mouth or eye tracking, and thus eye direction (both to monitor attention in relation to the dialogue and to help resolve a reference to an object in the scene); facial emotion recording; pointing gesture recording, especially those of the hand, and more general kinds of gesture made with the hands or the body. Moreover, we have already mentioned the force-feedback gesture in the case of a haptic interaction: this is a device that manages both the recording of the hand's position and the generation of a potential resistance toward the user. The point is to couple this device to an immersion in a virtual environment, the user seeing a graphical representation of his hand manipulating objects in the virtual scene. In this context, the force feedback makes complete sense: it simulates a touch perception that completes the visual perception.

There is no system that can carry all this out simultaneously and in real time, but it is an interesting challenge for the more technophile members of the HMD research community. We can see there are many possibilities and the computer challenges are vast: the processes matching these types of recording include many issues falling within the scope of artificial vision, signal processing, mathematical modeling adapted to the representation of configurations and trajectories, all that with constraints of execution speed, precision and abstraction in representations that the system can efficiently manipulate, so the system can confront these representations with those stemming from automatic utterance understanding. As (Bellalem and Romary, 1996) show us, for example, for gesture trajectories carried out on a touch screen, a representation of a gesture under the shape of a sequence of several hundreds of positions is simply unmanageable. It is necessary to abstract regularities and significant instants from it to reach, for example, a curve that can be described in four or five parameters. If this curve is then used to help resolve a reference to an object, it will be possible to confront it with a representation (also simplified) of the visual scene and the objects that appear in it.

Some processes require specific recording devices, with the immediate examples of a microphone for processing speech and of the keyboard for processing writing. Other processes can be carried out in various manners, from the most troublesome to the most transparent. An example of *troublesome recording* is the pointing glove that the user had to put on so the system can record the position and configuration of his hand or the glove with an exoskeleton required for force feedback. The increasingly common example of *transparent recording* is the camera or coupled camera system that allows the user the freedom to carry out various processes simultaneously, for example tracking his face and detecting the configuration of his hand.

Automatic speech recognition is a field in itself, and its use in HMD creates additional issues (Jurafsky and Martin, 2009). The idea is to go from an audio signal to a transcription according to a code which is more or less close to written language and requires various data sources, including the following: an acoustic model, a list of words in the given language, a dictionary of pronunciations and, the source of almost essential data to increase performances, a *language model*. This model is built from statistical corpus analyses. By bringing the notion of context (one, two or three previous words), it allows the system to calculate the probabilities and retain the most probable hypotheses for the word (or other unit) it is currently recognizing. In the framework of a speech dictation, the language model is built from calculations carried out on texts taken from literature or the written press. We maximize the size of these texts so as to refine the language modeling in terms of possible word sequences. Within an HMD system framework, building this model requires us to aptly choose the corpus used for statistical calculations: it is not necessarily relevant to only keep oral dialogue transcriptions, but having dialogues close to those expected by the system is a definite advantage, even if various language models have to be managed. Moreover, alternating the user and the system's interventions brings us an additional limitation: the probabilities for a user's utterance depend on what the system just said. The language models thus have to take into account the state of the dialogue, and become more and more difficult to manage.

There is an additional issue compared to vocal dictation: if the result consists of a written text matching what has been said, the speech recognition module result in an HMD system can be much more detailed. First, it can include various recognition hypotheses, so that the following modules make a choice depending on their own expectations. When an utterance includes an unknown word, i.e. a sequence of phonemes that do not match any of the words in the lexicon, the recognition module has a choice between various solutions: either bring it back to one of the words of the lexicon, even if the pronunciations are vastly different, or try to transcribe the sequence of phonemes with a potential spelling depending on the languages. While these two solutions might be acceptable for speech dictation, the second, for example, perfectly adapted to transcribing surnames that the system does not recognize, it is not the case for HMD: not only does the recognition module have to indicate that it is an unknown word, but it also has to transmit a code describing the word's pronunciation, so that the system can add it to its vocabulary and pronounce it in turn, if only to ask the user what it means. To get the job done, each recognized word is given a confidence score, and the syntactic or semantic analyzer uses these confidence scores and its own preferences to find (rather than have imposed) the most plausible transcription of the utterance.

An additional aspect with consequences on the nature of the result transmitted to the other modules of the HMD system is found in the prosody. Whether one is talking of the role of the recognition module or of another specific module, it is useful for

the written transcription of the utterance to be accompanied by coding, by a transcription of the prosody. We will see in Chapters 5, 6 and 7, that prosody helps in semantic analysis (by providing focalization clues), in solving references when a gesture is used jointly with a referring expression and in identifying speech acts, by providing a tone outline that allows us to privilege one hypothesis over another. We thus expect various indications from the prosodic analysis module: locating the focalization accents, temporal breakdown of the utterance, word by word, to match words and gestures in a multimodal dialogue, and a coding of the intonation's main characteristics. More in-depth analysis, with, for example, the detection of periods, requires additional indications, but for now falls more in the domain of subsequent oral corpus analysis than the domain of real-time analysis for the HMD. This is actually a criticism that can be used against many of the current systems: they do not use prosody, even though it is an essential component of oral language. Initiatives such as that of (Edlund *et al.*, 2005), who presents NAILON, an automatic prosody analysis system able to detect in real-time various prosodic characteristics of an utterance in HMD, are important.

One last aspect in which automatic speech recognition module has a role to play is speaking turn management. The HMD systems have long remained limited to an alternating operation of interventions, the system never interrupting the user and only starting to speak once the user has finished his utterance. More than that, we saw on page 31 with the push-to-talk button or pedal that it was on the user to let the machine know the beginning and the end of his intervention. We are now able to expect that an HMD system will let the user express himself at any point, with no constraints, and it is up to the machine to detect the beginning and the end of the interventions. This is actually one of the functions of the NAILON system, which uses prosodic clues of fundamental frequency and rhythm to automatically detect the end of a user's intervention.

1.2.2. *Analysis and reasoning abilities*

Once the signals have been received at the system's input and transcribed into appropriate representation, many analyses and reasonings will be carried out so that the system can understand the meaning of the user's utterance, his intent and, thus, the answer to give him. The analyses fall in the domain of automatic understanding of natural language, that is of NLP, and cover the following aspects: word identification (lexical analysis) so as to find their meaning (lexical semantics) stored in the system according to a well-defined formalism; the identification of the sentence's structure and the grammatical functions of the identified components (syntactic analysis); the construction of the sentence's semantics by combining the meaning of words and following their syntactic relations (propositional semantics); the allocation of referents to the first and second person pronouns, to referring expressions in general, and to anaphora in particular (pragmatic analysis sometimes called *first-level* pragmatics);

the identification of the implicit and the context attached to the utterance (*second-level* pragmatics); and the determination of speech acts so the system can understand the nature of the user's intervention (*third-level* pragmatics). Beyond the simple transcription of the *literal meaning* of an utterance, here we enter into the field of the determination of its *contextual meaning*. As for NLP, the implemented methods and algorithms have evolved for all these analyses. Where at one time, the symbolic approaches stemming from the AI were the only ones around, we now see statistical approaches, and they are at all levels of the list given above. These approaches have proved their efficiency in many fields of NLP, and have sometimes completely replaced symbolic approaches. In the HMD domain, it is the hybridization of symbolic and statistical approaches that provides us with the most promising results.

Starting with a semantic representation that is faithful to the utterance, we then reach an enriched representation through one or more implicit or explicit messages that the utterance carries forth. This enriched representation is what the system will confront, internally, to previously manipulated representation as the dialogue advances. It is also due to the information it contains that the system will be able to abstract the structure of the dialogue and compare it to structures that are considered for this task. This approach was imagined back in the 1990s (Reichman, 1985) but its computer-based implementation within a real HMD system framework only happened much later, and is still going on today. The system can thus carry out an assessment of the task's satisfaction, identify the deficiencies and decide what its next intervention will be. All this proceeds from the reasoning that it implements so as to process the user's utterance in the more relevant manner, taking into account what has already been done, what the utterance brings to the dialogue, and what still needs to be achieved to satisfy the task. Here we are situated not in linguistics and pragmatics but in modern AI: the themes approached are those of knowledge representation and especially following formalisms stemming from logic, in order to allow automatic deductions and those of expert systems and multicriteria decisions. Actually, as for the analyses mentioned in the previous paragraph, we are faced with a hybridization of various approaches. As an example, the approaches based on the expressive power of a well-defined logic, and its consistency with natural language, have explored different types of logics: propositional logics, modal logics, temporal logics, description logics and hybrid logics.

1.2.3. *System reaction types and their manifestation*

Once the system has decided what action to carry out as a reaction to the user's utterance, it still needs to materialize this action. In the case of a system which is only written or only oral, it must generate an utterance in natural language. Natural language generation is a research field in itself (Reiter and Dale, 2000) that includes various aspects such as sentence construction and the determination of referring expressions. We find here the same issues as those involved in automatic understanding, but in an inverted manner. Even if they have some linguistic resources in common,

the generation methods and algorithms are specific and are not a simple overturn of their understanding equivalents. In the case of an oral system, the last step carried out by the system is text to speech, that is pronouncing the chosen utterance. We can also find here the concerns of prosody: to look real, the utterance must be pronounced with intonation, rhythm and even focalization, and all these must be perfectly in keeping with the system's communication intent.

In the case of a multimodal system, for example when an avatar graphically represents the machine, the issue includes the gesture generation and their temporal synchronization with the words of the generated verbal message. When gesture is possible, the issues of generation, and especially that of determining referring expressions, reach a new dimension: speech and gesture complement each other, and the system must choose which part of the message to allocate to each aspect. Moreover, the graphical design of the avatar itself is a field that generates important questions about the realistic aspect of the avatar's physical appearance, its gaze and its movements: eye, eyebrow, lip movements when an utterance is verbalized, head movements, if nodding, and more generally body movements. The indications given to the user through these movements play a role in the human-machine communication, and it is essential for the various movements that have the same purpose, for example those of the eyes and the eyebrows that indicate the avatar's attention level, to be correctly synchronized. The emotions are also transmitted through gestures and are also a field of research on to themselves, which requires studies on the typologies of emotions, their relevance in HMD and the way they should be rendered, not only visually, but also in speech. Finally, in the case of a system including an HMI and manipulating a vast quantity of data, the issue also includes that of the graphical presentation of information (such as the IMMPS mentioned earlier). It can happen that the HMI itself generates earcons. The earcon generation and the speech generation must then be carried out in a relevant manner, i.e. without superimposing them, which might hinder the user's perception.

1.3. The current challenges

Can a machine think? The Turing test and Grail-style quest for the talking machine is still as fascinating as it ever was, but the current limits to the HMD systems mean that the question is not set in these terms any more. In a more pragmatic approach, the questions are set in terms of limits in the abilities of the machine to model and naturally process natural languages, to represent and reason on logical representations and to process and integrate various and varied signals. Commercialized versions prove this everyday: an HMD system only works well within a limited application framework, that is in a sufficiently limited framework for the maximum interaction possibilities to have been imagined upstream, during the design phase. Contrary to what attempts such as ELIZA had us believe, nothing is magical, and no matter what

technologies are implemented, whether or not they are symbolic, statistical, or involving machine learning or not. Everything must be anticipated, and this represents an amount of work proportional to the considered abilities for the system.

The main challenge of HMD remains, as it was in (Pierrel, 1987)'s time, the multidisciplinary design of comprehensive systems that allow the user to express himself spontaneously as he would with a human interlocutor, and this for a variety of applications, so as to offer systems that are accessible to anyone, in all everyday situations. More precisely, we will develop four sets of challenges: theoretical challenges, challenges concerning the span of expected abilities in a system, technical challenges concerning system design and technical challenges trying to help system development.

1.3.1. *Adapting and integrating existing theories*

According to (Cole, 1998, p. 191), recent strides have not included the development of new theories but focused on the extension and integration of existing theories. Thus, we can find many hybrid approaches that use the expressive power of more than one existing theory. This observation that we mentioned earlier when talking of the increasing closeness between symbolic approaches and statistical approaches is still true today. In linguistics, we mentioned the prosodic, lexical, syntactic, semantic, and pragmatic analyses; what was long considered to be a succession of analyses carried out one after the other is now approached in a completely different manner: a part of the results of the prosodic analysis is used for the semantic analysis and a part of the results of the syntactic analysis is used for the pragmatic analysis, and the latter is not monolithic but involves various aspects that are almost independent from each other. One challenge consists of completely reviewing the classic breakdown into natural language analysis levels, and better integrating the analyses that have common goals. In HMD, the goals are a list that depends on the targeted system but includes at least detecting the end of the user's utterance; representing its meaning in a logical manner, or at least, as a data structure that is usable by the considered algorithms; resolving the references to the objects managed by the applications; identifying the implicit content carried that has not been explicitly said by the utterance; updating the dialogue history; etc. Each goal in this list is reached due to the help and collaboration of various analyses. For example, to automatically detect the end of the user's utterance, we need a prosodic analysis that indicates when the tone outline dips and thus provides a hint, and we need a syntactic analysis, which shows if the sequence of words captured until then is a grammatical sentence or not, and whether or not is needs additional words. Depending on the system's personality, especially its tendency to interrupt the user, we can even imagine that a semantic analysis provides an additional argument, as soon as a semantic result is obtained. If we remain within a cascading analysis operation, this type of mechanism is impossible. One of the challenges is thus to explore the collaborative analysis implementations. If we start the first analysis at the end of the user's utterance, then we lose any possibility for the system to interact in real time.

One of the challenges is thus to carry out analyses at all times, almost one for each word uttered by the user. If we consider a module to be a black box that gives a result at one time and within a single data structure, then the prosodic analysis should not materialize itself in a single module but in various modules: one for the determination of the tone outline, one for the detection of prominences, one for the rhythm, etc. A modular breakdown that follows the breakdown into linguistic analysis levels cannot be justified any more, and the application of linguistic theories to HMD is still the focus of research. In multimodal dialogue, the integration of theories is all the more crucial: the gestures are linked to prosodic aspects, ergonomic aspects, etc. As we will see in Chapter 2, collaboration between fields is essential.

Finally, to end this list of theoretical challenges, let us underline the importance of methodology, with which to carry out experimentations and to create and use reference corpus for the HMD. This challenge is linked to resources and is key not only for the study of dialogues covering a specific task, but also to carry out machine learning algorithms or to generate data such as lexicons, grammars and language models for the oral dialogue as well as the multimodal dialogue. In this case, one of the challenges is in a better integration of these resources. As an example, the OZONE project we have mentioned allowed us to reflect on the concept of meta-grammar (or meta-model), with the goal of instancing from a joint base of linguistic grammar and statistical language model.

1.3.2. *Diversifying systems' abilities*

The technical challenges linked to the abilities of HMD systems are the NLP, AI, ECA, QAS and HMI tasks, and many more. In general, all the components we have mentioned could be improved, with a greater scope of phenomena taken into account and a greater finesse in their processing. (Cole, 1998) highlights various linguistic aspects such as exploring the nature of discourse segments and the discourse relations, as well as the need for additional mechanisms to manage key phenomena such as the highlighting of information in a linguistic message. All these challenges focus on the same goal: increasing the coverage, the fluidity and the realistic aspect of the dialogue. To make it more clear, the goal might be to achieve a natural dialogue system or even a natural multilogue system in natural language (Knott and Vlugter, 2008), which will be multimodal, multilingual, multitasking, multi-roled, multi-thread, multi-user and, of course, capable of learning...

The question of *realism* is a great question, which starts with speed: a system taking 10 seconds to answer does not have a chance of achieving realism. If this criterion can be measured, however, there are some that cannot: how should we measure the realism of a synthetic voice, of sentence construction, of an ECA's gestures? The fact that some users reject an artificial voice is sometimes based on tiny details that are hard to measure, such as a minute defect in elocution rhythm. The perception of these

minute defects can create unease and disturb the user. The field of robotics or that of computer-generated images use the term of *uncanny valley* to describe this type of phenomenon. The issue is that we try and get closer to the human (to reality for computer-generated images) but there is still a small gap between what is achieved and what is aimed for. And this gap, as minute as it might be, is enough to be perceived, and to irritate. To counter this, some designers make the gap visible and forego the goal of getting close to the human. So some mechanical toys that look like animals do not have any fur. In HMD, for example, the Web service ANANOVA takes on the appearance of a gorgeous young lady... with green hair (Harris, 2005, p. 341).

Finally, a key challenge for the abilities of an HMD system is its robustness, that is its ability to manage its own shortfalls, at a linguistic analysis level for example, its own deficiencies and errors, and its ability to always bounce back, to help the dialogue progress no matter what the cost, by using the task to be solved, or not. This implies the design of modules able to operate with incomplete entries and have strategies to manage problems. This also implies to predict, from the first stages of design, tests and settings with real data, real conditions, rather than laboratory-controlled conditions.

1.3.3. *Rationalizing the design*

At the design level, there are multiple methodological and technical challenges. Once the list of understanding and generation abilities is determined, they have to be instanced and organized into modules, components or agents in an architecture. The interaction languages between these elements, the evaluation methods, the construction methods of necessary resources and the integration methods have to be specified. The main challenge here is the rationalization of the architectures' engineering (see Chapter 4), and in general the rationalization of production flows, as in any professional technical field. Thus, (Harris, 2004) focuses, Chapter 9, on a very precise description of a design team, with the different professions involved: a dialogue team leader; an interaction *architect*; a lexicographer in charge of the aspects linked to corpus; a *scriptwriter* in charge of anticipating the expected types of dialogues, but also the definition of the system's personality and its possible reactions; a *quality engineer*, without forgetting the ergonomics experts, technical experts as well as an expert in the field covered by the task. The task, and more generally the context of the dialogue, can require integration into another field of research. A first example is robotics, in which we are starting to see systems integration abilities belonging to robotics and HMD abilities, preferentially in a multimodal manner (Gorostiza and Salichs, 2011). A second example is that of HMI when we try to give them speech, while keeping, on the one hand, the possibilities of directly manipulating the HMI and, on the other hand, the HMI advantages in terms of ergonomics, plasticity: adapting to the user, the terminal, the environment.

1.3.4. *Facilitating the implementation*

At the level of system development, the technical challenges are found in the facilitation of development processes. A first step on this path is the multiplication of toolkits devoted to HMD. VoiceXML is a basic example, but there are many other platforms devoted, for example, to helping design multimodal dialogue systems (López-Cózar Delgado and Araki, 2005). A second step would be the implementation of a library offering a rich and performing panel of NLP tools and dialogue managers. This is an important challenge and was tentatively introduced by products such as APACHE's OpenNLP for some aspect of written NLP. An *OpenDial* library would probably be useful and would help focus efforts elsewhere rather than on the components that all systems have in common. Finally, a third step in the same direction would be the materialization of a whole set of services linked to vocal recognition, text to speech, prosodic, syntactic and semantic analyses in a software layer such as middleware, or better yet, in a computer extension card, such as graphics cards for 3D visualization. This challenge, if it happens someday, would allow it to be exceptionally easy to develop a system: all processes would be carried out in hardware rather than software, which increases the speed, and it would really open the door to systems usable in real time. Obviously, this is not a simple challenge, and if we compare it to 3D, for which the graphics card works much more during the design than the final product, which needs overspecific and overdelicate processes, we could imagine that a *dialogue card*, at first, would accelerate and simplify the design of systems without carrying out the full development.

1.4. Conclusion

The quest for a machine able to understand human language and answer its user as well as a human interlocutor has gone on for more than 50 years. The issues arising from natural language processing have not allowed us to achieve completely comprehensive systems yet. However, we can note a diversification of communication modes and aspects of the human-machine interaction. By relying on the theoretical, methodological and technical stages that have marked the history of the human-machine dialogue, this chapter has outlined the abilities considered for a human-machine dialogue system, and the current scientific limits and challenges in this field.

Chapter 2

Human-machine dialogue fields

The term *dialogue* refers to the different types of interaction between two people and characterizes a certain use of language. (Clark, 1996, p. 23) presents six proposals covering the use of language: it has a social goal; it is a joint action; it involves the meaning desired by the speaker and interpreted by the hearer; it often involves more than one activity; its study is both a cognitive and a social science. These proposals allow us to start identifying the fields involved in dialogue studies: not only linguistics and pragmatics, of course, but also social sciences and cognitive sciences. We will discuss these aspects in this chapter, with a goal of finding application possibilities of HMD for these studies.

At the end of Chapter 1, we mentioned a list of professions that could create a work team to design an HMD system. This list came from (Harris, 2004) and mostly puts emphasis on computer-related professions. We could very well imagine the team to include linguists, sociologists and psychologists. We can also, or mostly, imagine that it includes specialists of the touching zones between these fields, especially researchers positioning themselves between linguistics and computer science, and who help develop systems by extracting and formalizing linguistic theories and models, which they believe can be applied to HMD. The application of a linguistic theory to HMD can require its simplification, or at least its transformation so as to obtain a slightly different linguistic theory which is nonetheless able to be directly implemented and thus tested. This is what happened for most of the systems mentioned in the previous chapter, especially those that arose from the speech act theory (Austin, 1962; Searle, 1969) and the dialogue hierarchical structure (Roulet *et al.*, 1985). In general, the approach tries to formalize the interpretation of natural language and the management of the dialogue is as follows: it starts with the linguistic study of (written and oral) communication phenomena, their methodical analysis, classification and characterization,

i.e. the identification of their relevant aspects. This phase requires general linguistics and sociology, which provides additional explanations to the functioning of speech turns and polite forms of address. Then comes the modeling phase, which consists of identifying the rules that allow us to take into account the maximum amount of phenomena. This phase requires formal linguistics, NLP, and computer science as soon as we reach formalization and implementation. It comes with hypotheses, to determine rules left unclear by the theories, and then borrows experimental protocols from psycholinguistics to test these hypotheses. Finally, the dialogue management and message generation to the user involve decision making for which psychology and cognitive sciences are crucial. For example, it is by taking into account concepts of cognitive load and work memory that the system can adapt itself to the human user's abilities.

This chapter, which we can approach as a list of the fields we should know before starting in the HMD field, draws a summary of the cognitive aspects (section 2.1), linguistic aspects (section 2.2), and computer science aspects (section 2.3), the latter becoming numerous as the system processing abilities get bigger.

2.1. Cognitive aspects

An HMD system is an example of a *cognitive system*, i.e. a system characterized by the facts that it processes outside data, that is has knowledge and that its behavior is based both on this knowledge and the processed data. Depending on the way the cognitive system is modeled, we can follow the structural cognitive science path, which highlights structures and mechanisms of structure operation, or the path of computational cognitive science, which highlight the processing of an information flow and tends to assimilate humans, as cognitive systems, with artificial cognitive systems.

Cognitive sciences cover the fields that study natural or artificial cognitive systems. In fact, they are also the fields that focus on language as a human faculty. Linguistics is obviously a part of this: it approaches language faculties through the study of language as a system of signs, with a semantic aspect that focuses on meaning and is thus close to semiotics whose more specific object is the meaning of signs. Psychology is also a part of this: it focuses on the behavior of humans and approaches language through the thought operations involved in language activity (cognitive psychology), child's acquisition (developmental psychology) or even through the influence of other individuals (social psychology). Sociology studies the behavior of groups and societies, and approaches language through the study of interactions, as already mentioned with the conversational analysis approach and the study of collective representations carried by language. Neurosciences, which can be compared to clinical psychology and psychopathology, study the brain's anatomy and functioning, and more specifically approach language through its pathologies. Psycholinguistics, a branch of cognitive

psychology obviously linked to linguistics, and also historically linked to information theory with notions of information, code and message, studies the psychological activities through which a subject acquires and implements a language system, notably through laboratory experiments. Neuropsycholinguistics brings the focuses of psycholinguistics and neurolinguistics (a branch of neurosciences covering the understanding, generation and acquisition of language) together and focuses on the study of the activation of areas of the brain.

As for philosophy, with its aspects of language philosophy and more recently cognitive philosophy, it contributes different points of view, including history of ideas and the logic underpinning language, and philosophy provides epistemological clarification. Just like cognitive philosophy, the *cognitive* qualifier allows us to specify and even outline a new scientific field in an established discipline. This is, for example, the case of cognitive ergonomics, which refers to the study of interactions between a human user and a device that thus involves mental charge notions or human factors and that is involved in the design of HMI and HMD systems, especially for automatic generation and multimedia information presentation. This is also the case for cognitive linguistics which, at first, provides linguistics with the notion of cognitive plausibility, and has now established itself in a disciplinary field around questions on links between language and though, on the nature of knowledge constituting language faculties, or the computer-based modeling of knowledge.

Moreover, the AI, which we have already discussed, is historically one of the cognitive science fields, after the attempts with the first cybernetics (general study of the mind's operation, involving as much psychologists and anthropologists as mathematicians and logicians), the second cybernetics (study of the cognitive systems and self-organization evolution), then the system theory that emerged with robotics and expert systems. Its goal is to develop artificial systems, and it approaches language through computer-based modeling and simulation (Garbay and Kayser, 2011). Even though we now speak of ambient intelligence much more, this is of course the path of HMD. (Guyomard *et al.*, 1993-2006) show how AI provides an additional point of view to linguistic approaches: while they suppose that the structure of the dialogue focuses on the utterances themselves, their shape and linguistic content, the AI, by associating itself with logic and language philosophy, provides notions of planning, representation, reasoning and speech act to explain the dialogue structure. It also contributed in defining a plan-by-plan approach of HMD, also called differential approach. In this section, we will explore the cognitive science points of view and their potential applications in HMD around the abilities of perception, representation and learning.

2.1.1. *Perception, attention and memory*

The human cognitive system has perceptive mechanisms, which allow it to distinguish, and thus extract, objects from their environment. These mechanisms are fast,

automatic and reliable. They are characterized by certain remarkable aspects such as invariance: in visual perception, an object is always recognized, even if it is rotated or if the scale is changed (Gaonac'h, 2006). Audio perception and spoken language perception also involve specific mechanisms, which are worth noting from a cognitive point of view, such as the ability to select a sound source in a noisy environment, for example. Another example, the essential characteristics of the perception of spoken language are intensity (loud or soft voice), pitch (high or low voice) and timbre as a set of aspects such as texture that lead to the speaker being identified. Visual perception and audio perception have been the subject of an impressive number of experiments and models in cognitive sciences. The Gestalt theory, the Adaptive Control of Thought (ACT) model by J. Anderson and those derived from it, or even more specialized models such as those focusing on the perception of speech and lexical access (Forster, Morten, Marslen-Wilson models), were repeatedly presented in literature, see (Gaonac'h, 2006). However, their application to HMD are very few, and this is the point that we will focus on.

The HMD system is a cognitive system which does not aim to have the same abilities as a human cognitive system. On the one hand, it is not necessary to reproduce the complexity of the mechanisms involved, for example in recognizing a word, when an adapted computer technique, speech recognition, reaches the same results on its own. Moreover, reproducing the perception mistakes, weaknesses and limitations of the human is not desirable: the lack of performance of the automatic analyses is more than sufficient for that. Our inability to watch several visual scenes at the same time, or listen to an audio message in the right ear and a different message in the left ear, is not useful in defining the processing abilities of an HMD system. However, they are results that a system can take into account when it creates output messages; these messages are meant for a human, they should thus not involve two simultaneous visual scenes or two simultaneous audio messages. This is the matter of limitations which we will explore in Chapter 9 under the name of *human factors*. The perception abilities to be implemented vary from one system to another. If it is useful to give a robot visual perception so that it can move within an environment, the question is not raised in the same manner for a system such as SHRDLU or OZONE. Indeed, in the first case, the robot does not know the objects placed in the environment in advance; it thus has to recognize them, otherwise it cannot move itself. *Artificial vision* techniques are thus put into use. In the second case, the visual scene shared is managed by the system; the system displays the objects, so it knows them and knows their disposition. In other words, the distinction mechanisms of an object in its environment do not come into play in SHRDLU and OZONE. This does not mean that all the visual perception mechanisms should be ignored. On the contrary, knowing where and how the objects are displayed on the screen does not mean knowing how the user will perceive this scene. One of the objects could be visually salient for a human, according to criteria such as size, color, shape; criteria that will cause this effect of visual salience in a predictable manner. If we provide the system with abilities that will automatically

detect this salience, or even with abilities to automatically detect perceptive groups of objects, then we increase the system's understanding abilities. It will now be able to approach the scene not only from its point of view as a machine, with each object's absolute position, but also from a point of view that is closer to human cognition (Landragin, 2004). The point is that with this, the system can understand and provide compensation for the user's biased or incorrect perception.

It is the same thing with attention phenomena. The cognitive sciences have clarified the different forms and functions of attention (Gaonac'h, 2006, p. 137): attention retention (reacting to new things and maintaining awareness); attention selection (filtering certain types of information); attention allocation (managing simultaneous activities); attention control (controlling the progress of actions, i.e. preparing, alternating, supervising). Providing an HMD with such abilities and limits only makes sense if you are trying to reproduce human fallibility. If on the contrary, you are trying to design an understanding, performing system able to use any clue that allows it to solve the task for which it was designed, there is nothing to prevent all the recorded information to be processed, without selecting or allocating the system's *attention*. However, in this case as well, the human attention characterization is a result that the system can use when it generates a message for the user.

Memorizing and forgetting give rise to different questions. In the models stemming from cognitive sciences, various types of memory are distinguished. At the level of immediate perception and the preliminary mental representations, we talk of very short-term or short-term memory. This is the level at which the *memory span* comes into play, a threshold limiting the number of elements stored in this memory, with the consequence of preventing a human from rendering a greater number of elements. At the level of the first mental activities, we talk of long-term memory and, for example, distinguish between semantic memory, which stores general, verbalizable knowledge, from episodic memory, which stores specific events that happened at a given time and left a mark in the user's mind. In HMD, taking into account that memory span is a strategy for knowing the user's limits so that he is not overwhelmed with information, for example during the message generation phase. It is thus applied in the same manner as previously mentioned. However, the nature of semantic and episodic memories can also provide ideas during the system's design itself: it is, for example, a way for remembering to save any specific information during a dialogue, such as past events; it is also a way for separating the different sources of knowledge in the system's architecture, and to specify the encoding processes and recorded information retrieval. More than that, the notion of forgetting can be interesting: after a certain amount of time or a certain amount of speech turns, we can consider that a piece of information that was transmitted and not retrieved is forgotten, or at least labelled as such, so that the system, if needed, can convince the user to provide this information again.

2.1.2. *Representation and reasoning*

Once the words have been perceived and recognized, the human cognitive system has mechanisms to process them, i.e. identify their meanings, represent them mentally, link the meanings one with another, until it has built meaning for the sentence and utterances and the representation of the knowledge transmitted. Cognitive sciences have suggested various models for each of these questions. According to the authors – once again, see, for example, the works presented in (Gaonac'h, 2006) – the representation of a word's meaning is reduced to a finite set of elementary characteristics that may be linked to a prototypical representative of the class designed by the word, or to a list of implications, of propositional content, of procedures applied in context, provoking meaning effects in certain conditions. Depending on the approaches, linking the meanings of words with one another can involve a network of relations between representations, or more complex models based on aspect or proposition structures. Some authors, such as G. Fauconnier, consider language and its use to be a mental construction of spaces and elements in these spaces, with links between the spaces. The term *mental representation* itself is used in the model by (Reboul and Moeschler, 1998, Chapter 6), the theory of mental representations, which describes, with an approach falling within the field of cognitive pragmatics, a set of elements, relations and operations on multiple input structures modeling mental representations, for the resolution of references. The understanding of language is the object of various works in cognitive sciences and linguistics. As we will see in section 2.2, the models mostly come from linguistics, and the contribution of fields such as cognitive psychology relies on experiments into models that take into account human limits during controlled tasks such as reading texts, or again in cognitive plausibility judgments on models built in linguistics. Cognitive sciences have contributed to underline the importance of context and clarify the nature of knowledge which takes part in building representations: general knowledge about the world, about individuals, about physics systems, about objects, about their categories, their properties; knowledge about actions, their conditions, prerequisites and results; knowledge about languages; formats or patterns ready to be used to build a representation by providing a structured framework with models such as those of scripts and plans.

The same questions can be asked when designing an artificial cognitive system, and the models stemming from cognitive sciences provide interesting paths for computer implementations. On the one hand, the structures of mental representations and different sources of knowledge are almost directly implementable: classical computer data structures match finite sets of characteristics and relations between structures. The management of propositions and implications is less self-evident, but is still within reach of computer science. On the other hand, cognitive sciences themselves learn about implementation possibilities as means to carry out tests. This is the case for certain approaches exploring the notions of script or plan to determine the succession of actions allowing us to carry out a task and to use the resulting structures when understanding a text or managing a dialogue.

Once the knowledge has been mentally represented, the human cognitive system has mechanisms to reason on this knowledge and build new data by inference. The way to infer a new piece of content from premises characterizes the type of reasoning involved. Depending on the approach, we thus distinguish the reasoning by deduction whose conclusion is as certain as the premises, in the way of "X is Y. Every Y is Z. So X is Z"; the reasoning by induction generates general representations from specific facts, and can thus lead to mistakes, for example, after meeting so many gray cats, we conclude that all cats are gray; the reasoning by analogy uses a similarity considered not to be random between two pieces of knowledge. We can also mention reasoning by abduction, which deletes improbable solutions. All these types of knowledge manipulation are implemented, whether by using an existing inference engine, which might even be the base of a type of computer language, for example, in the case of the famous PROLOG, or by building specific inference mechanisms for a specific artificial cognitive system. The choice of the type of inferences to be taken into account in the development of an HMD system has consequences on the way to interpreting a user's utterance and the decision the system will take depending on the nature of the inferences carried out.

The human cognitive system makes decisions not only based on mentally represented knowledge but also depending on its relations with other humans: each of our decisions is the result of a calculation in terms of beliefs, desires and intentions. Determining the nature and role of these mental states is also one of the tasks of cognitive sciences, and is also applied in HMD. A human cognitive system is able to reason on the basis of its own mental states and is able to allocate mental states to its interlocutors. It is the approach, among others, of J. Searle on intentionality, or the principle of the interpreter strategy developed by D. Dennett, and the mind theory that is linked to it (Reboul and Moeschler, 1998, Chapter 9). In computer science, such cognitive theories have led to the specification of rational communicating agents defined by a set of formalized mental states. In the BDI model mentioned in paragraph 1.1.2, three mental states are modeled:

– beliefs, which correspond to the level of trust in a piece of datum, this piece of datum becoming knowledge when the belief is true;

– desires, which correspond to a set of possibilities available to the cognitive system given his beliefs and knowledge;

– intentions, which are a subset of desires: those that have been selected and lead to an action.

This model inspired many researchers and led to extensions, such as the BOID model, which also takes into account the speaker's obligations (Broersen *et al.*, 2005). Finally, to finish with this brief and altogether too schematic overview of cognitive mechanisms, let us emphasize that the derivation of many inferences can lead to a multiplication of beliefs and knowledge, and that, with this excessive combination,

notions such as *relevance* allow us to select and only keep the most important, significant or relevant data depending on the term kept by relevance theory (Sperber and Wilson, 1995). Such criteria operate within human cognitive systems and also allow us to control reasoning mechanisms in artificial systems. Some authors add a more general ability of *metacognition*, that is a cognitive mechanism focusing on the cognitive mechanisms presented above and allowing the system, be it human or artificial (Sabah, 1997), to have an idea of the processes it is implementing, so as to optimize these processes, for example.

2.1.3. *Learning*

Human learning can happen explicitly through education, or implicitly through association, analogy or even action and exploration (Gaonac'h, 2006). When it comes to knowledge, the goal is to memorize new pieces of knowledge, in order to acquire, create or delete links between already acquired knowledge, to create or modify categories, or to build new scripts or plans. Learning thus concerns all the fields mentioned above, from the moment when knowledge is mentally represented. In human dialogue, cognitive sciences have described various situations that involve learning. Language is a privileged vector for any type of knowledge transmission, and the dialogue between an expert and a learner is the very first corpus. Depending on how the dialogue goes, we can distinguish active learning, during which the learner asks questions and in some ways directs the dialogue according to his desires, and passive learning, during which the expert directs the dialogue and transfers knowledge. The language itself is the focus of learning, and the dialogue between an adult and a child is a second corpus of study, as is the dialogue between an expert and a learner trying to acquire a second language. (Clark, 2009) thus presents a list of phenomena characterizing the dialogue between an adult and a child. First, the child tends to mispronounce words, which leads the adult to over-pronounce them. The child makes morphological mistakes, for example when conjugating verbs, lexical mistakes and syntactic mistakes. This leads the adult to correct him, either when the child has finished talking, or by interrupting him. At the dialogue structure level, the main observation is that the child does not follow the same structure as that which can be observed in a dialogue between two adults: the child is often self-centered and tends to follow his own trail of thought rather than interact with the adult. Moreover, the adult tends to prefer closed questions, whose answers are yes or no, to open questions, whose answers require expressing knowledge. He creates repetitions and rephrasing, to increase the chances of information transmission. He also adapts himself to the words that the child knows, so he aligns himself with the child from a lexical point of view and follows a chronological, or at least logical order, when telling a story, for example.

These learning situations and their characteristics can inspire HMD system designers, without the system or the user being considered as a learner. In situations where the system has to give the user an explanation, if only to have him wait while carrying

out the query, providing information in a chronological manner can help the generation and in the same way, generating closed questions can also do so. If the machine is able to *learn online*, that is to enrich its linguistic abilities during the dialogue and through it, we can plan for specific processes for the user's utterances with the aim of correcting or of assessing the achieved progress.

However, human learning and machine learning are very different, and a machine does not have to go through all the stages a child does. In the case of language learning, if it is decided to provide the machine with this ability, then the learning does not focus on all the dimensions of the language: pronunciation and morphology are much less problematic than lexical semantics or syntax. The HMD does not focus on *simple* mistakes, and there is a chance that, from the moment when morphological rules are correctly defined, there is no learning to be done on this aspect again. Beyond the specific case of online learning of linguistic abilities, most of the systems in question learn implicitly. For example, if a user shows negative behavior translating his displeasure with the machine, it should remember the conditions which triggered this behavior to avoid replicating it. As negative behavior happens more often in similar situations, the machine will confirm its caution with regard to those conditions. This is the principle of *reinforcement learning*, a type of learning based on experience, and is one of the challenges currently explored by HMD: the learning abilities allow the machine to avoid difficulties such as anticipation and the programming of every possible situation. By following this principle to the extreme, we reach the principle of *supervised learning*, that is controlled by an operator, which answers the questions being asked by the system and allows it to learn reliably. Such a procedure happens before the real use of the system, during the design phase. This is prior learning, which can be carried out following a similar dialogue to that planned for the user, or more directly, by the operator intervening in each module in question. The data can then be provided step by step, as in a dialogue, or all at once, by using a learning corpus, in which case we talk of batch learning. In any case, the goal is to train the system to refine its behavior, or simply to optimize its performances (without learning anything new), before it is made public.

From a technical point of view, machine learning can take various shapes depending on the type of data and algorithms in question, see Chapter 6 of (Garbay and Kayser, 2011). A historical distinction separates symbolic learning stemming from AI, which is limited to symbolic or at least discretized data, and numerical learning, linked to statistics. In the 1990s, with the rise of the digital age that we mentioned in paragraph 1.1.2, numerical learning takes precedence over symbolic learning by using the strength of calculation and great size of the corpora. Today, as I. Tellier writes it in the preface to (Tellier and Steedman, 2009), the work consists of formulating the problem that requires learning first, either through categorizing, or labelling (or annotating), and then choosing the suitable technique, knowing that the support vector machines (SVMs) are performing in categorizing and that the conditional random fields (CRFs), with the hidden Markov models (HMMs) are performing for labeling.

At the HMD level, we find all the approaches mentioned, and in a variety of modules. Some aspects, such as automatic speech recognition, have long used statistics and numerical learning. What we have seen more recently is a rise in hybrid approaches, which accommodates both symbolic and numerical learning, for all the modules in an HMD system. The NLP modules largely follow this approach (Tellier and Steedman, 2009), as do modules falling within the field of pragmatics. Thus, the resolution of object reference now calls on statistics (Funakoshi *et al.*, 2012), as well as on semantic and discursive representation models (Stone and Lascarides, 2010), on the common ground and grounding process (Rossignol *et al.*, 2010), the speaker intention recognition (de Ruiter and Cummins, 2012), the allocation of mental states, the dialogue management and automatic answer generation (Rieser and Lemon, 2011). We can see that one cannot make an impasse on machine learning in HMD anymore.

2.2. Linguistic aspects

An HMD system manipulates language; it is thus mainly concerned by linguistics in general, and notably by automatic linguistics, also known as NLP but not exclusively: the recent advances in the corpus linguistics and what goes with it, computational linguistics have changed our way of analyzing human dialogues. We have defined the word *corpus* as a collection of averred linguistic material. More precisely, a corpus is built following strict rules, so as to obtain a sample of the language chosen according to explicit criteria. Depending on the type of dialogue, it can be audio or written samples. For the oral dialogue, studying recordings has never been very practical, and the corpus is often enriched with a transcription, i.e. a textual approximation of pronounced utterances. The resulting text is sometimes hard to understand without a transcription of the prosody, for example pauses, and it is needed to add codes or annotations, i.e. data related to textual units, for example words. We then have an annotated corpus, which has both sentences and observations on their uttering. When you carry out morphological and syntactic analyses, it is now usual to annotate the corpus with a set of morphosyntactic labels and syntactic trees, or more simply, relations between words and groups of words. In the end, a corpus can take the shape of a computational database with many fields and multiple exploration and interrogation possibilities. For that is the whole point of corpora: to create a set of analyses so as to carry out frequency calculations, more complex *descriptive statistics* such as factorial correspondence analyses, correlation research between text and annotations, or even *inferential statistics* such as variance analysis, and in that case the goal is to try and generalize the results obtained from a specific corpus. This disgression on corpus linguistics allows us to add statistics to the list of disciplines involved in HMD, as a tool for analyzing dialogue corpora.

2.2.1. *Levels of language analysis*

All the fields of linguistics and all its schools can inspire HMD: from the moment when a model is suggested, it is tempting to apply it to HMD and have it undergo the test of computer implementation. Thus, the fields of phonology, morphology, syntax, prosody, semantics and the field of pragmatics, more or less close to linguistics, provide models that hold potential interest for HMD. Indeed, some of these fields sparked many more applications than others, and other fields such as the study of oral language, the study of dialogues with conversational analysis and discourse analysis approaches have sparked much interest, through the focus of their studies, of course, but also because the multi-disciplinarian exchanges were fruitful, with spin-offs on each side. Before we develop all the linguistic and pragmatic aspects involved in automatic understanding (Chapters 5, 6 and 7), this section aims to provide a few prerequisite numbers concerning the study of the oral language and of dialogue.

The oral language that we can observe in a spontaneous dialogue does not look like the written language in literature or the written press: there are many hesitations and repetitions, and the accumulations of terms aiming to find the correct word through trial and error are also common. The oral utterance is built in real time, whereas the author of a written text has had all the time in the world to find his words and refine his sentences. These differences lead to two points of view in relation to syntax: either oral and written language are characterized by their own specific syntax; or share the same syntax, with compromises, for example a slackening of constraints for oral language. These two points of view are applied to HMD. In the first case, we study the syntax of oral language to specify the system's syntactic module. In the second case, we add a set of additional detection and repair mechanisms for oral language specificities to the syntactic module based on written language. Moreover, the same words do not appear with the same frequency in the oral and written languages, which has consequences on the language models useful for automatic speech recognition. As an example, (Blanche-Benveniste, 2010) notices that adjectives are 25% of written words and 2% of oral words, or that the words "of which" ("dont" in French) are used with a few dozen verbs in the written language, but almost exclusively with the verb "to speak" ("parler" in French) in the oral language. For an HMD, we can thus draw a predefined list of the uses of "of which" ("dont"), which will simplify the preparation of linguistic resources, at least on that point. In a general manner, (Blanche-Benveniste, 2010, p. 11) shows that if the frequencies (and uses) differ, it is indeed a single syntactic framework, which allows us to account for the oral or written language. The main difference is in fact in morphology: written language involves spelling corrections, which are rich in well-known rules and exceptions, whereas oral language involves a specific morphology, focuses, for example, on the pronunciation of connections between words. As for syntax, this observation has important consequences on the design of HMD systems. In certain oral systems, the speech recognition module produces a written transcription of the user's utterances, and the transcription is then processed by a morphosyntactic analyzer. The latter relies on the

morphology of the written language, and that of the oral language is simply ignored. In most cases, it is taken into account in the transcription itself, since pronunciation helped determine this transcription. But in other cases, the written transcription introduces ambiguity, such as with the French word "plus" which can mean "more" or "no more" but which is never ambiguous when spoken, since the meanings have different pronunciations. Ideally, either the morphosyntactic analysis is carried out directly on the oral language, or annotations are added, describing the pronunciation to the transcription, or back-and-forth motions between the modules are implemented, with, in this case, an analysis module detecting ambiguity and requesting the recognition module to solve it without starting on a fresh transcription and analysis.

In a dialogue, the language is dependent on speech turns and interactive aspects. A written sentence is not interrupted, or it is deliberately so, whereas interrupting an interlocutor happens during a dialogue, whether it is oral or written. Moreover, the dialogue is characterized by opening and closing utterances, that is key-moments in which the messages follow predefined codes. These are greetings, farewells, polite forms of address, or excuses. Finally, a dialogue is studied as a relevant succession of utterances, which means there are valid reasons for an utterance, "here are your possible itineraries" to follow the previous utterance, erI would like to go to Paris. We can refer to these reasons as coherence, thematic continuity, or reactive action following an initiating action. In any case, a dialogue is analyzed as a set of utterances, which some refer to as a discourse. (Cole, 1998, p. 198) reminds us that research on HMD has historically followed two paths drawn by research on human dialogue: first, the discourse analysis path, from speech acts theory (Searle, 1969), which sees dialogue as a rational cooperation and thus draws links between an act and the previous one (see paragraph 1.1.2), and, second, the conversational analysis path, which approaches dialogue as a social interaction with organization phenomena such as speech turns, abrupt topic changes and disfluence (Sacks *et al.*, 1974). The discourse analysis path has led to computational theories of speech acts, and the conversational analysis path has helped improve speech turn management in HMD. For example, (Sacks *et al.*, 1974) show that speech turns are regulated, at least in American language, by rules applied to specific moments, called transition-relevance place (TRP), that is moments in which the language structure allows for interruption. The criteria can be prosodic, such as the presence of a pause, syntactic or semantic: we can consider that a sentence is potentially over when the agents required for the predicate have been expressed, even if the speaker wishes to add precisions, for example a deferred complement, a repetition or a rephrasing. This example is important in HMD because provides parameters for the module in charge of detecting the end of the user's interventions, a module that is involved in systems able to manage real-time dialogues. In the same way, we can test the possibility of interrupting the user. There are two arguments that might subdue this initiative: first, even if the interlocutors theoretically have the

opportunity of speaking at the same time, (Levinson, 1983) shows that all the superimposition zones only reach 5% in the human dialogues he has studied; second, HMD does not necessarily need to imitate human dialogue.

This last point was the focus of a more general debate on the management of speech turns. We have suggested an approach aiming to allow the user to express himself in his everyday language and to allow him to do so spontaneously. This approach, tending toward natural dialogue in natural language, relies more or less on the premise that HMD can copy human dialogue. Actually, any HMD system has limits, and the user quickly realizes that the machine has understanding limits. He adapts his behavior and utterances, and thus contributes to a dialogue which takes a completely different direction than it would have with a human interlocutor. Depending on the HMD system's abilities, this adaptation can be moderate, and thus similar to the adaptation that any interlocutor requires, or major, to the point of considering that the machine is a very specific interlocutor. This is the point of view of (Jönsson and Dählback, 1988), in an article with a very striking title: Talking to a Computer is not like Talking to your Best Friend (paragraph 3.2.1). This is also the point of view of (Fraser and Gilbert, 1991) who suggests the Wizard of Oz methodology to turn the type of communication between a human being and a machine into a subject of experiment (paragraph 3.2.3).

2.2.2. *Automatic processing*

The aspects of NLP linked to automatic understanding (Jurafsky and Martin, 2009) and automatic generation (Reiter and Dale, 2000) are obviously involved in HMD, on lexical, syntactic, semantic and pragmatic aspects. Thus, dictionaries and digital resources, semantic lexicons or even lexicalized grammars provide content and methods to store lexical knowledge (Mariani *et al.*, 2000). Syntactic formalism, especially when it has been designed with computational concerns in mind, allows us to design efficient performance analyzers, which are open to additional knowledge (Abeillé, 2007). The semantic models, from the conceptual graphs (Sowa, 1984) and up until the formalisms of formal semantics and discourse semantics (Kamp and Reyle, 1993), allow us to combine the contents of words and utterances (Enjalbert, 2005). Pragmatics, with the resolution of reference in a purely linguistic or multimodal context (Pineda and Garza, 2000), with the identification of the implicit, such as presuppositions (Van Deemter and Kibble, 2002) and that of speech acts (Traum, 2000), has also been the focus of various works in NLP which are directly applicable to HMD.

Other fields of NLP are involved in a more occasional manner, or for a certain type of dialogue. The resolution of anaphora, which is an aspect of the resolution of references, is also a field unto itself, with its own algorithms and assessment campaigns (Mitkov, 2002). An HMD system does not generally need to implement the most complex algorithms, inasmuch as, in a dialogue, an anaphora tends to refer to a recent, or even immediately accessible, antecedent. However, the integration of

algorithms that have proved their worth in their respective fields is obviously beneficial, if the machine's resources allow it. Another field of NLP, the identification of coreference chains can be applied to HMD and thus provide additional clarification, for example on the way the user introduces a new referent and then refers to it; the system can thus reproduce this behavior at the generation level. The identification of discourse relations (Asher and Lascarides, 2003), or the detection of named entities are also NLP applications useful in HMD, especially for open domain dialogue.

NLP has not solved all the issues and its limits are visible in HMD. It is the case of the cover and subtlety of the language, which it cannot record efficiently and reliably. This is the case when identifying ambiguities, when managing unknown words, when identifying a non-literal use of the language: deadpan, irony, sarcasm, exaggeration, rhetorical questions (Clark, 1996, p. 353). For these questions, HMD systems implement local procedures, which depend on the task and operate without providing any kind of general solution.

2.3. Computer aspects

The computer aspects involved in designing an HMD system are numerous, as the history of the field has shown, notably the processing of an audio signal, the processing of recorded gestures through a specific device, face tracking and more generally artificial vision, NLP, knowledge representation, inference engines, machine learning, text to speech synthesis, multimedia information presentation, or even the annotation and statistic study of corpus. All these goals require theoretical proficiency, be it linguistic, psychological, ergonomic, logic, mathematical, and involve specific data structures and algorithms. In Chapter 4, we will explore a different field of computer science, that is software architecture and model-driven engineering. We can see that the panel of technical skills necessary to implement an exhaustive system is extremely vast. As in the previous section, here we will very quickly present a few landmarks on two aspects chosen randomly: digital resources and plastic HMI.

2.3.1. *Data structures and digital resources*

The era of the 14 KB database of the PARRY system is long gone. Today, we seek to maximize the amount of data recorded in the system, to provide the maximum amount of material to analysis algorithms. The achievements publicized by IBM, such as Watson, are proof of this; the data and brute calculation strength allow us to achieve operational systems. This, at least, is one of the ways to achieve it, as the reflection on data is still essential. In NLP and in general, with the rise of the semantic Web, efforts have been made for years to improve the quality of the data. This goes from using standardized languages to adding metadata to data that describe the content in a sober and universal manner, or to ontologies, set structures of terms or concepts common

to a field or a task. In HMD, we kept a closed operation for a long time, with data internal to the systems and not connected to the Web. On the contrary to resources available on the Web, which are in constant evolution and can reach unreasonable sizes, these partitioned data have the advantage of being very quickly accessible, in real time, an aspect imposed by spontaneous HMD. The data, however, require being first specified, and one of the challenges of current HMD systems, and especially open-domain systems, is to use the Web's resources in real time.

2.3.2. *Human-machine interfaces, plastic interfaces and ergonomics*

ECA, robot companions and voice-activated HMI explore new ways to interact with a machine, which are not solely based on language. The visual or sound-based transmission of emotions plays an important role in ECA, for example. As for the HMI on which a voice-activated module was grafted, they are usable with the traditional means of menus, keys, cursors and other graphical metaphors. Voice-based interaction and traditional interaction can thus mix and make the interaction management and dialogue management more complicated. For example, does a cancellation command cover the last action carried out or the last action carried out with the same means of interaction? This is the type of question asked with the integration of voice command to our cell phones and personal digital assistants (PDA). Moreover, and this is also a field of research that is involved at the moment, going from a desktop computer to a laptop through a PDA raises the question of the adaptation of the HMI to size constraints and of the use of hardware on which it is run. For the user to be able to switch hardware transparently, *plastic interfaces* have to be designed, i.e. real-time adaptive interfaces which will not stop the application from running. The issue is particularly significant when voice-activated interaction and even multimodal interaction are involved (Kolski, 2010, Chapter 9). In this case again, it is a challenge for the future HMD systems.

2.4. Conclusion

To design a system able to interact in a realistic manner with a human user, one approach is to take inspiration from human operations, and thus linguistics when it comes to language use, cognitive psychology for the perception or attention aspects, and in general, cognitive sciences. Whether we are trying to simulate a human communicating mechanism, to simplify it to specify a computer module either able to approach it or take inspiration from it, there are various scientific fields involved. This chapter has given us details of some of these contributions, and finished with all the computer science fields concerned with the design of a dialogue system, fields that are becoming increasingly numerous.

Chapter 3

The development stages of a dialogue system

Once the system's goal has been defined, for example to provide the user with information on train timetables, and the interaction means, vocal or multimodal, have been chosen, the designer(s) face a variety of questions concerning, on the one hand, the phenomena that the system absolutely has to deal with, and on the other hand, the development stages, identifying the system abilities, dividing the system into modules, determining and building the required resources for each module and, of course, software development. An HMD system is a piece of software like any other, and thus it follows the classic design stages of specifications, through which we define what the system should do; development, during which we define how the specifications are implemented; and then assessments, to ensure that the system we have obtained is usable and matches the specifications (McTear, 2004). These stages do not necessarily occur in a linear manner, with no possibility of returning. Thus, we can develop a first system, assess it and depending on the results obtained, develop a second more efficient system. Thus, (Jurafsky and Martin, 2009, p. 872) highlight three main phases with potential back-and-forth movements between them: an initial phase of studying the user, for example using interviews, and of studying the task, for example with similar HMD systems or human dialogues; second, an implementation phase, which can cover simulators and prototypes rather than the targeted system directly; and third, assessments, especially iterative tests with users. Depending on whether the cycle concerns a real system, a prototype or a simulator, the tests can take many shapes, especially if we add the possibility of involving a computational simulation of a user rather than a human user.

The choice between the different possibilities mentioned above is based on technical and methodological criteria. We present these aspects in this chapter, with section

3.1 illustrating some representative study cases, and section 3.2 detailing the methodologies involved in the main design stages.

3.1. Comparing a few development progresses

The following scenarios are purely indications; even if they are based on real experiences, they tend to caricature them slightly and their main purpose is to illustrate methodological concepts.

3.1.1. *A scenario matching the 1980s*

In the 1980s, an HMD system could be developed by a single person who had a vast knowledge in computer science. The designer imagined here is trying to test a theoretical dialogue model, which is, for example, the innovative proposal of his dissertation research.

To design his system, he defines a chain architecture, with modules that are executed in a chain fashion, in the following order: automatic speech recognition, linguistic analyses, solving references, speech act detection, natural language generation and text to speech synthesis. To implement this system, he works on each module one after the other, following the machine's processing order. He uses an existing speech recognition module, sets it to have a vocabulary matching the tasks he is aiming for and trains it with his own voice for a few hours in order to optimize the performances. To operate, this module needs a button or a pedal to be pressed when the utterance is pronounced. Once the utterance is finished, which is indicated by releasing the pedal, the module starts to process the recorded signal. It creates a written transcription of the pronounced utterance, which enters a linguistic analysis module. This analysis, using unification grammar, for example, includes lexical, syntactic and semantic aspects: in such a grammar, the lexicon, the grammar rules and the sentence representation use the same formalism, i.e. feature structures. Thus, the same feature structure unification mechanisms allow us to carry out different stages of automatic understanding. The designer writes the resources that the module requires. The semantic representation obtained is enriched by the reference resolution module that uses a database containing all the application objects, so the complete representation thus achieved can be used as a base for the identification of the language act characterizing the utterance. Thus, we obtain a representation made of an illocutionary value (order or question, for example) in a semantic context. This representation enters into the module concerned with dialogue management, which implements the designer's model: depending on the utterance's illocutionary value, the system will choose its answer. The generation is simple, uses patterns, that is sentences with gaps in them, which are predefined according to the categories matching the different possible illocutionary values. For each of the modules mentioned, the designer develops his own local

algorithm from the available data: its definition of the task, of the vocabulary and of the matching linguistic phenomena. For text to speech as for automatic recognition, he uses an existing piece of software. All the modules carry out their tasks on the same computer. The speech recognition and text to speech modules are demanding when it comes to memory and machine calculation time. There are thus a few seconds between the end of the user's utterance and the beginning of the system's response.

The designer can then test his system by taking up the progress imagined for a few typical dialogues on the selected task. Clearly, the results are not good. He has to take up the settings, or even the development, of each module again. To refine the understanding abilities, he rewrites part of the resources. After a while, he starts the tests again, playing the role of the user like he did during the development phase. No metric assessment is used. Intuitively, and also because it would be a shame to throw the system out after so much effort (especially since nothing can be reused for a different system), the results seem to have improved. The designer realizes, however, that all the words defined in the lexicon have not been used: on the contrary, the system can only understand and process a small subset of the lexicon. No matter, it is now possible to dialogue with the system. Even if the resulting communication is rather simplistic, it has the advantage to be possible.

3.1.2. *A scenario matching the 2000s*

In the 2000s, various designers came together within the framework of an international project or a perennial laboratory activity sustained by constant human means. After a joint specification phase, they divided the experiment, development and assessment work. The first stage of experiments consists of getting test subjects, who are not the designers, in front of a system simulation as it is imagined, according to the Wizard of Oz principle. This step that ends with the subjects being interviewed can be completed with human dialogue corpus studies, to help define the essential aspect and behavior limits of the system, and thus of the Wizard of Oz, for the given task.

The development starts by the joint definition of the system's architecture, and the specification of the behavior of the modules constituting this architecture: input types, output types and module communication language. Each of these aspects draws strongly on the state of the art, by using, for example, a communication language established in another project. By following such constraints, we foster both the integration possibilities of the existing modules and the reusability of the components created for this specific project. The designers then start the development, i.e. building resources and coding algorithms. To achieve this, the order followed is not that of the processes carried out by the system; on the contrary, we start with the system's core, dialogue management and pragmatic aspects. By simulating the input, we then refine the system's behavior, and then specify the understanding abilities required for this behavior. We can then implement the modules that carry out linguistic analyses.

The resources are partly taken from other projects and partly from available, free or licensed, resources so as to keep the maximum efforts for the algorithms, or rather for their settings, taking the task into account.

The next step consists of testing and assessing the already implemented embryonic system, by simulating the input. Then, just as the dialogue management abilities have turned into settings for the design of understanding modules (as well as generation modules, since it follows a similar development), we move on to the materialization of the language correctly processed into parameters for the voice recognition and low level processing. An existing automatic recognition system is used, with much better performances than the system used in the previous scenario, and increased interaction possibilities. As for each phase of development, tests and assessments are immediately carried out with just the recognition and then all of the system, so as to refine the coverage of the resources and the process's settings. The assessments mostly consist of measuring the recall and precision level, a classic method for all the NLP systems (Chaudiron, 2004), adapted to the separate assessment of modules but not the system as a whole.

Finally, the system obtained undergoes user tests, i.e. subjects are once more involved, who are neither the designers nor the original Wizard of Oz subjects, and their opinions are collected. They are analyzed by the designers, who then decide what improvements should be made, and might carry out a new development phase: improving the core of the system, then the understanding and generation modules and finally the voice recognition and text to speech.

This scenario covers the weak points of the scenario discussed in paragraph 3.1.1. However, even with a rational architecture, most of the processes are chain processes, and the essential aspects of the oral language such as prosody are ignored. This is typically the point when we consider the issues described previously on the consequences of not taking the oral language's morphology into account.

3.1.3. *A scenario today*

A design scenario today would have to provide answers for all the weak points mentioned previously, and all the issues of section 1.3. Without discussing them again, let us note that a system's development can now require not just the existing modules, but a toolkit that provides a completely configurable general framework. As for any framework, it could become penalizing due to the limits it imposes. In this case, the development of additional modules is crucial, and is the focus of previous experiments, tests and assessments through metrics that are more precise than recall and precision calculations. The HMD system itself can now be globally assessed with methods that calculate the task's completion level, on indexes such as the number of speech turns to reach the originally defined goal. We will discuss these aspects in

Chapter 10. Moreover, all the development stages are being rationalized, so that nothing is done blindly or by chance. For example, the corpuses are made for each of the identified modules, and these corpuses are themselves broken up into various parts: one part for training (machine learning is integrated into the stages that need it most), and one part for tests. Another example, the Wizard of Oz, is carried out with additional constraints and precautions. This is the type of constraints we will now detail, going over a few emblematic design stages.

3.2. Description of the main stages of development

3.2.1. *Specifying the system's task and roles*

An HMD system is usually used to help the user in a given task, whether it is train ticket reservations (closed domain) or general information (open domain). The system thus has the role of managing the dialogue along a path that should quickly lead to the task's completion.

Compared to an HMI or a classic Website, an HMD system gives the user the freedom to express himself as he wishes, and calls him in a natural dialogue in natural language without having overstrict guidelines. The system's role is thus to manage the dialogue with the specificities of human language and dialogue.

Are these two roles compatible? Human dialogue studies have identified a certain number of aspects to cover the dialogue's natural aspects, 7 for J. Sinclair, 9 for (Warren, 2006), or even 10 according to (Clark, 1996). These aspects highlight the fact that the dialogue is interactive, cooperative and consistent to the point of sometimes being predictable, and that its success relies on the two interlocutors. (Clark, 1996, p. 9) criteria include the following: copresence of the participants in a same physical environment, visibility, audibility, interaction instantaneity, evanescence (the utterances are fleeting, as well as the interlocutor's actions), and also simultaneity and the real-time aspects of the interaction (the interlocutors can process and generate at the same time). These criteria provide great principles on natural dialogue, both on the interaction conditions and the dialogue's progression. By underlining cooperative aspects, an HMD is thus seen as a partner rather than a tool, and we can deduce that the task's resolution is a joint activity between the two interlocutors and not controlled by the system.

However, a task such as a train ticket reservation follows structured principles. The system needs to know a precise set of information: the departure station, the arrival station, the date, the times, the class (first or second) and various other preferences. More than that, the order in which this information is given by the user follows certain principles, as (Luzzati, 1995, p. 91) shows: "a time slot request which does not indicate the arrival station and departure station is inconceivable, whereas it can be

considered without any time reference: everyone knows that you should always get the train line before looking for the train time". The dialogue's progression is thus strongly limited (Kolski, 2010). Moreover, studies of the SNCF corpus show that the vocabulary is relatively limited, as are the sentence structures, which means that the task's predominance influences the natural language.

The main question is thus as follows: when there is a task, does the task's resolution have precedence over the dialogue's spontaneity? Either we consider that the dialogue is first finalized, and the answer is yes, even if there is a lack of consistency (efficiency above all); or we consider that the dialogue primarily aims to maintain enjoyable communication with the user, and accept that it might take three or four more speech turns to arrive at the same result. Both choices are acceptable, but must not be assessed in the same way: if the speed in satisfying the task drives the assessment, the finalized system will obviously arrive first.

This question takes on a particular dimension when we question the notion of spontaneity. There is not necessarily a gap between the task's resolution and natural dialogue in natural language. The natural aspect of the dialogue is not subsequently judged by analyzing the lexical breadth and the linguistic and pragmatic phenomena diversity, but is judged on the one hand by the *user experience*, with the answers given during an interview on the ease of dialogue with the system, and the level of satisfaction given by the system's utterances, and on the other hand, by the subsequent analyses checking that the system's reactions are relevant to the user's utterances. A user can be satisfied by the system's dialogue even if the task took longer to complete than planned. As (Sperber and Wilson, 1995) and later (Reboul and Moeschler, 1998) show, relevance is at the heart of the natural dialogue in natural language.

The question between solving the task and natural dialogue is also linked to the machine's interlocutor role. As we saw at the end of paragraph 2.2.1, talking to a machine is not the same thing as talking to your best friend. Unless he is misled and believes, as it happens over the phone, that the interlocutor is human, the user knows he is talking to a machine, which can lead to a specific type of behavior on his part. The Wizard of Oz experiments and user tests at the end of the design are clear on this point. After a Wizard of Oz test on a train ticket reservation task, (Luzzati, 1995) shows that the dialogues focus on the key points with simple structures: each initiating act is matched by a reacting act, the lexicon is essentially that of the task, that is reduced, and there is no particular digression. The users do no argue their decisions, and do not have to defend them. They do not make any comment to explain what they are saying. As for references, they do not make any reference to themselves or to the machine. In other words, the dialogue remains natural but is focused on itself, which means the task is satisfied. In such conditions, we can consider that it is indeed a natural dialogue in natural language. Since the user is talking (or thinks he is talking) to a system, he limits the breadth of his utterances, but not at the cost of spontaneity, since he does it naturally. If we took this to the extreme, this mechanism

could lead to a caricature operating mode, observed, for example, during a Wizard of Oz described in (Landragin, 2004), in which the user limits himself to two or three sentences: those that worked at the very beginning of the dialogue, and during which the user gained confidence. This is the type of behavior that can be found with a genuine HMD system, and that leads to the question, again, of the natural aspect: if the user limits himself so greatly, then talking to a machine must be disturbing him. He is not in a natural mode. Actually, as the experiments carried out in the MIAMM project show (Landragin, 2004), some users limit themselves without being prompted, and others place themselves automatically in a natural dialogue in natural language without any suspicion against the system. However, and this is one of the essential challenges of the designers, the system has to hold a conversation: if, on the contrary, it starts showing signs of not understanding, the most confident user will change his behavior very quickly.

3.2.2. *Specifying covered phenomena*

There is often a loop made up of various stages when imagining the understanding and dialogue abilities of a system: reflecting on the system's expected behavior, which means a specification of the interaction's nature; specification of the breadth of the system's abilities; a simulation of the future system and thus the constitution of a dialogue corpus; a corpus study, that is an analysis of the expected and new phenomena; reflections on how to take these new phenomena into account, and thus a return to the first stage. Once the loop is stable, we then move on to the design of a processing model, which takes a number of processes into account, and this model is then implemented and tested to find its weak points. When the implementation has been carried out, we can then come back to the experiment stage and start the loop again, by using a real system instead of a simulation.

During the specification of the nature of interaction, the question of the possible communication modalities between the human being and the machine arises, as well as the question of the recording and generating devices involved. If the dialogue happens over the phone, the only modality is oral, at the system's input and output. If the dialogue happens face to face, that is on a computer with potentially an avatar displayed on screen, the input can be written, oral or multimodal, with, for example, gestures made on a touch screen or with a mouse. Through a desire for consistency, and so as to not disturb the user by using a different modality from his, the output can then occur through the same modalities, or at least equivalent modalities, and the gestures carried out by the system can materialize through a display on the screen and not a specific device, except in a robotic system.

The specification stage for the system's breadth of abilities involves three additional methods: the imagination of dialogue situations, the carrying out of experiments such as system simulations, the analysis of dialogue corpus to deduce phenomena and

situations that have to be taken into account. For each of these three means, we can add a phase consisting of spreading the phenomena identified by a set of similar phenomena, i.e. derive new ideas from observations. This *derivation* operation ("he said this when he could have said that, so both have to be taken into account") allows us to achieve a more satisfying set of phenomena in terms of size and coverage.

3.2.3. *Carrying out experiments and corpus studies*

The Wizard of Oz experiments we have mentioned and the dialogue corpus studies complement each other. These two methods allow us to use the concrete dialogues as bases for the system's design. Any experiment, if it is recorded, is also a corpus that can be studied in the same way as the corpus coming from human dialogue recordings, human-machine dialogues (with no simulation or deceit), computer-mediated human dialogues (same comment) and even machine-machine dialogues as when we have two HMD systems that interact with each other.

The Wizard of Oz, abbreviated to WOZ, named after a character by F. Baum, or also PNAMBIC (Pay No Attention to the Man BehInd the Curtain) thus consists of simulating an HMD system for a human being, the wizard or fellow, to observe the behavior of a subject faced with this simulation (Fraser and Gilbert, 1991). The subject believes he is talking or writing to a machine, but the latter is linked to another computer controlled by the wizard, who can answer in written or oral form, by generating his own utterances or choosing from some predefined patterns. If the simulation is correctly carried out, the subject does not discover the deception and adopts the behavior he would have with a machine, which already allows us to analyze this type of behavior. The dialogues are recorded to allow the designers to detect problematic situations, the cases in which the wizard's reactions disturbed the subject and obviously the cases of misunderstandings. By focusing on the incriminated modules, the designers can thus improve their system and make it more robust. To detect the moments where a subject is disturbed in a reliable manner, and to detect the moments of incomprehension or those of hesitation, additional techniques are implemented at the same time as the dialogue recording: we can, for example, track the subject's face, so as to record his facial emotions or use an eye-tracker to detect his eye movements and deduce the observations on his attention and, when there is a shared visual scene, on the objects observed, for example, when solving a reference.

For a Wizard of Oz to be usable, it requires the experimental conditions to be well defined. However, there are as many ways to design a Wizard of Oz as there are system: the wizard can be one of the designers, which allows him to generate a behavior close to the one of the targeted system, but he can also be a second experiment subject, who does not know the goal of the experiment and only applies a set of rules aiming to simulate the system (double-blind principle, also known as *ghost in the machine*). Moreover, the subject and the wizard's messages go through a computer system, and

the latter can integrate a specific process so as to complicate the experiment. In (Rieser and Lemon, 2011), noise is added to the subject's utterances before transmitting them to the wizard, which disturbs the dialogue and allows us to increase the frequency of clarification requests. In this case, one word was randomly deleted, but we could very well imagine that a word would be replaced by another or by other local heuristics, and not just for the subject's utterances but also for the wizard. Obviously, it is easier to carry this out in a written dialogue than in an oral dialogue. In the case of the very complex methodology implemented in (Rieser and Lemon, 2011), the Wizard of Oz's goal is fourfold: to observe dialogue situations; to create a study corpus and thus model a machine learning model; to create a learning corpus; and to contribute to a user model specification, that is a computer program whose goal is to simulate the behavior of a system's user. Each step and goal is accompanied with precautions, assessments and confrontations with other methods such as supervised learning. In general, the directions given to the wizard can be more or less precise. If they leave him free to react as he wishes to the subject's utterances, there is a risk of obtaining a more fluid and robust dialogue than an HMD. These are the main criticisms that we made against this type of experiment. (Cole, 1998, p. 199) thus states that the Wizards of Oz are often overoptimistic when it comes to the performances of the targeted system, which leads to the design of systems that lack robustness. (Rosset, 2008) adds that when it comes to choosing among a set of predefined possibilities, a wizard is necessarily slower than a system, which degrades the natural aspect of the dialogue and prevents the use of the recordings that have been carried out. (Denis, 2008, p. 90), who focuses on cases of misunderstanding, shows that in dialogues obtained by Wizard of Oz systems, the misunderstandings do not last as long as in HMD: even when he simulates a misunderstanding, the fellow always manages to re-establish an understandable dialogue in the utterance that follows the subject's detection. Unless the fellow is trained on this specific aspect, the Wizard of Oz does not allow us to specify reliable repair strategies for the HMD.

For a corpus to be usable in a relevant manner, we have to take into account the conditions in which it was obtained, that is in what type of dialogue situation it took place and, if it is made of extracts, according to which criteria the extracts were selected. In any case, the corpus never reflects the possibilities of natural dialogue in natural language: it is not complete, it is only a *reservoir of phenomena*. Depending on the conditions and selection criteria, we can consider this reservoir to be more or less representative. As a corpus is considered increasingly representative of a certain dialogue situation, so we can increasingly use it to design a system to communicate in the same situation. When the corpus reaches a sufficient size, this use can comprise frequency analyses: we implement the most frequent phenomena in the corpus as a priority, or at least we try to optimize their processing. The frequency of a phenomenon's appearance however does not have anything to do with its importance: utterances such as "warning!" or "stop!", which can happen in the command of systems

in dangerous environments or when managing air traffic control, are very infrequent, but should not, however, be neglected by the understanding module.

Moreover, a corpus can be used as a lasting reference in HMD, and even NLP, fields. This is the case when particular attention is given to the transcriptions and coding of extralinguistic aspects: coding prosody, face and hand gestures and in general the video showing the user mid-utterance. The AMITIÉS system, for example, has involved the annotation at various levels of a corpus consisting of not only the transcription of the speech, but also the identification of the speakers, the marking of the zones in which speech is superimposed as well as semantic, dialogic, thematic and emotion annotations (Hardy *et al.*, 2006). The efforts made are such that, as for the SNCF corpus used by (Luzzati, 1995), various HMD systems can use them.

3.2.4. *Specifying the processing processes*

The data almost determine the processes that have to be implemented in order to process themselves. The more the chosen phenomena are diversified, the more the processes will have to be deepened. As an increasing number of ambiguities appear, so the processes will increasingly rely on fine linguistic, multimodal and dialogic analyses as well as an appropriate management of context. This section illustrates the methodological difficulties that appear at that point, by taking as an example a designer who we imagine facing the system he is designing alone, a bit like in paragraph 3.1.1. Our designer determines the system's architecture, develops or reuses each of the identified modules, each of the necessary resources, faces the difficulties he comes across and carries out a certain number of simplifications. And actually, it is by implementation that difficulties appear and initial ambitions diminish slightly.

Let us consider, for example, the initial phenomena including anaphora, essentially pronominal anaphora but also a few examples of associative anaphora. The reference resolution module, which brings together the resolution of direct references, demonstrative references (and thus the multimodal fusion of pointing gestures with linguistic referring expressions), person deictics and anaphora, must thus integrate a pronominal and associative anaphora resolution system. The designer thus first turns toward existing anaphora resolution systems (Mitkov, 2002). If we take into account the implementation context (specific issues for the French language, a considered lexicon, a panel of phenomena, input and output formats), he quickly realizes that adapting, setting and integrating an existing resolution system into his architecture is a delicate matter. He then decides to implement his own anaphora resolution algorithm directly into the module with the relevant settings. However this task is only one of the aspects of one of the system's many modules, he has to ignore certain phenomena, such as all the associative anaphora, to achieve an operational resolution system in time and with reasonable means. It is a diminishing factor (and hard to admit), but it happens.

As another example of simplification linked to multimodal dialogue, let us cite the processing of combined multimodal references such as a single demonstrative gesture linked to two or three referring expressions, or several gestures linked to the same referring expression (or to several referring expressions but without matching each to its own). It is very difficult to design a multimodal fusion module able to identify these situations, and it quickly becomes time-consuming with the inevitable technical problems such as detecting the beginning and end of a multimodal utterance, the recording of the gesture's trajectory and the temporal synchronization management. So as to develop a system able to have reaction times close to human reaction times, the low level input processing must be very fast and avoid complex management of all the gestures – referring expressions matching hypotheses at the end of utterances. Thus, and this is the case for many systems, we tend to forget about this type of situation and focus on demonstrative references involving a single gesture and a single referring expression, which is already complicated enough (see Chapter 6). In the end, the system only processes a subset of the phenomena identified at first, but at least it works. Of course, in the case of system designed by a full team of developers, or in the case of an existing system being reused, the problems do not appear in the same way. In those two cases, the process specification of the processing is directly linked to the system architecture's definition (see Chapter 4).

3.2.5. *Resource writing and development*

The real issues appear during implementation, as it usually is in NLP. When programming the main algorithm of a module, we notice that there are not enough resources, that carrying out the process is more complex than predicted and that it is necessary to reduce the number of phenomena processed. We might also notice that there is a missing input data for the module, for example a prosodic aspect that we had neglected but that ends up an important setting at a given moment. We could also notice that the module's output will not be as comprehensive or precise as planned. We could notice that the algorithm execution is slower than expected (or hoped). And finally, the implementation creates no surprises for designers who are not at their first HMD. In any case, the clarity of specification, the verification of available resources and the exchange between various specialists are classic solutions that may not be original but are essential in designing the system.

Is there an order in which to develop the modules? It appears at first that if the input and output of each module was correctly defined and remained stable, the order does not matter and the development can be split among various people. In reality, as we showed this with the scenarios in section 3.1, it is better to start with the system's core, that is the dialogue manager, and end with the surrounding modules such as speech recognition and text to speech. Taking this precaution will provide more tolerance for the specification's little mistakes and to avoid surprises during the development phase.

Is there a list indicating resources? It obviously all depends on the processing abilities considered by the system, and the possibilities of online or offline retrieval of existing resources. With what we have already presented, as well as a few additional elements that can be deduced and that we will describe later, here is an indicative list that shows the diversity of models involved in an HMD system:

– Domain models: database of instanced objects, potentially visible and referable, for example SHRDLU cubes and pyramids, and the physical model describing potential behaviors and evolutions of these objects.

– HMI models: in the case of an HMD with a supplementary interface displayed on screen, an operation model of the latter (actions–reactions and rules of priority of HMI commands over voice commands).

– Design models: ontology, graph or tree of concepts in question, with both objects and actions that can be carried out on these objects.

– Task models: sequence of actions leading to the task's completion, conditions and decision tree.

– Acoustic-phonetic models: for speech recognition, as well as for text to speech (they are not the same models but can have some parts in common).

– Prosodic models: tone outlines, rules for putting in relief, for analysis as well as generation (same comment).

– Symbolic linguistic models: lexicon, grammars, semantic structures, types of logical forms, rules for enriching these logics, deduction or induction mechanisms, etc.

– Statistical linguistic models: language models to help recognize speech, statistics on utterance structures, on speech act detection, on utterance sequences, etc.

– Gesture models: generation rules and gesture shapes to point at an object, to speak, to transmit emotions, which can be applied in analysis just as well as in generation in the case of an avatar display.

– Models linked to specific modalities: face shape bases for face tracking and analyzing modules (emotion models, for example), lip-reading models, writing recognition models.

– Cognitive models: human limits to take into account when generating message, cognitive charge, attention and salience management.

– Dialogue models: possible structures, general conventions, polite turns of speech and corresponding reactions, dialogue opening and closing patterns, speech turn passage rules.

– Interaction models: beyond the natural language dialogue, management models for interaction that can be oral, visual, multimodal or via HMI.

3.2.6. *Assessment and scalability*

In a development cycle as we have seen in HMI, like a *V cycle* (Grislin and Kolski, 1996), the tests and the assessments are not only carried out at the end of the design but also during various stages: the development of modules leads to unitary tests, then the integration of modules in global architecture leads to integration tests, which then lead to global system tests, the latter allowing us to check that the operational specifications have been followed. Finally, acceptability tests are implemented to validate the needs analysis. We will explore the different types of tests to be considered in HMD in Chapter 10, and we will focus here on two specific methodological aspects: the scalability and re-implementations.

Scalability is about going from the scale of a correctly assessed laboratory prototype to a genuine operational system usable by the public at large. The following are the main challenges:

– Going from controlled laboratory conditions to real, variable and uncontrollable conditions.

– Going from limited and highly reliable resources and data to real data, of great size with potential errors, noise and non-homogeneity.

– Going from a system's occasional execution for a relatively short period of time each time to a continuous operational mode involving a minimum of rebooting.

– Going from a user mode allowing a certain margin of error and dysfunction to a use that supports no dysfunction tolerance and has a very limited tolerance to mistakes.

– Going from a user mode involving a user both framed and willing (often an expert) to a mode involving a demanding and sometimes malicious user, always ready to play with the system or even to try and break it by any means possible. There is nothing more distressing for the designer than to watch such a user, who is refered to as the final user, torturing his system.

The notion of *re-implementation* comes into play after the assessment. It consists of modifying the system's code so that it can process the few problems it has identified as the most common or the most problematic. The goal is to minimize this phase of code updation, and do anything so that at the beginning of design, it has to be the easiest and fastest activity in the world. (Rieser and Lemon, 2011, p. 16) underline that today, there is no rational method in HMD to turn assessment results into code. This is however the case in other computer science fields, for example computer graphics and image synthesis (the notion of relighting after the full calculation of a rendering), and this is again an essential methodological challenge for the years to come.

3.3. Conclusion

When we start developing a dialogue system, we do not start to program directly and let the issues crop up as we go. Instead, anticipation is crucial and many stages of work are carried out before any kind of computational development: defining a task that matches the goals that the system must fulfill, observing and analyzing the human dialogues focusing on such a task, determining the linguistic and interaction phenomena that the system must process, carrying out experiments, specifying the system's software architecture, etc. This chapter provides examples of development scenarios including these stages before being implemented in development, testing and assessment.

Chapter 4

Reusable system architectures

The term *architecture* has various meanings, even in HMD. It can refer to conceptual architecture, i.e. all the software components (modules) and communication means between these components. It can be software architecture, i.e. the materialization of the conceptual architecture as a computational solution, for example a multi-agent system.

Moreover, architecture can refer to the system's organization as it runs, thus reflecting its operation. We then talk of *run-time architecture*, a conceptual architecture describing how the final system is built. First, it is a diagram of modules, with a specification for each of input, output and processes run, and second a diagram of the communications between modules. Architecture can refer not just to the running but also the designing of the system, i.e. not the system itself, but the system that enabled us to create it. We then talk in that case of *design-time architecture*, which may (or may not) happen during the design phase. More specifically, it is the development system's conceptual architecture that created the HMD system. This is essentially a set of sequential or parallel constraints, which creates a software chain to help development by automatically deriving certain resources from other resources, and automatically generating resources or even modules from models. A toolkit is the software materialization of a design-time architecture.

Once these definitions have been given, we can make various observations compared to the past and current achievements in HMD. First observation: conceptual run-time architecture is probably the most important characteristic of a system. It allows us to describe it and to show what its characteristics are. The conceptual run-time architecture is a kind of leaflet that highlights the system's innovative aspects and that proves it is not a quick improvement on an old existing system. There are multiple consequences that lead to two other observations. Second observation: despite the

wish to reuse existing resources, each system is defined by its own run-time architecture. It is directly linked to processing, understanding and generation abilities, and since all systems do not run with the same inputs and output, it is thus logical that the architectures are different. However, the current techniques allow us much more flexibility than before as to the operation of partially implemented architectures, and we should be able to rely on reference architectures more easily. It remains that the proposal of a new system is often accompanied by new architecture, with the leaflet and innovative effect mentioned earlier. Third observation: whereas the specification of an architecture should help collaboration between researchers, it is, on the contrary, sometimes a source of problems. Each researcher has his own suggestion, and conciliating approaches can be a long and delicate process. More than that, it happens that some people have a tendency to try and include other people's proposals as modules in their own architecture. What comes from these issues is that there is a lack of generic, reliable and trustworthy reference architecture that can be applied to any system. The rise in design-time architectures, if it is confirmed, might help give new solutions to this problem.

This is what we will discuss in this chapter, with sections 4.1 and 4.2 on run-time architectures and design-time architectures, respectively.

4.1. Run-time architectures

4.1.1. *A list of modules and resources*

(López-Cózar Delgado and Araki, 2005, p. 5) show us how a very complete run-time architecture, which notably integrates modules to process specific modalities such as lip-reading or tracking the user's face. In general, all the books mentioned in the introduction present architectures. These are generally diagrams made up of boxes (modules) that are labeled (processes) with arrows between certain boxes (processing sequence) that can themselves be labeled (type of data exchanged).

In the 1980s, the architectures often followed the order of processing that we have already mentioned: speech recognition, syntactic analysis, semantic analysis, dialogue management, automatic generation and text to speech. The variations often focus on the way to manage data. Thus, (Pierrel, 1987) presents a set of static data and dynamic data. The first set covers a subset of models that we have listed at the end of Chapter 3. The second set can be reduced to the dialogue history and the user model, essentially turned toward useful voice recognition settings: individual acoustic models, settings on the way to pronounce links between words or on prosodic outlines. For some tasks, the user model also includes access rights and control rights to the application objects.

Today, we find in most of the systems some equivalent to these modules, with more modules devoted to a specific modality or the management of a specific device. But as

(Cole, 1998, p. 198) states, the dialogue manager is always at the heart of the system. The dialogue manager manages the dialogue history, which records the occurrence of speech turns as the dialogue progresses, the utterances pronounced, their linguistic characteristics, especially the referring expressions used and referents mentioned, so as to find the necessary information when solving a new reference, a new anaphora or a nominal or verbal ellipsis. Moreover, the dialogue history also stores the task's state, the stage achieved in the current dialogue strategy or a description of the successes and failures in communicating.

More than a resource affected to a module, as was the case in the 1980s, the dialogue history from now on can be a fully-fledged module, with access and storage procedures. Indeed, any kind of process, such as syntactic analysis, can theoretically call upon it. Our comments in paragraph 2.1.1 on how to apply a forgetting process in HMD go in this direction. This principle of access and storage at any point is generalized in the current systems, especially if we try and get closer to real-time operations, that is with analyses carried out during user utterances. Thus, the module in charge of recording the audio signal stores the signal in real time is an *utterance* resource, and, still in real time, the speech recognition modules and prosodic analysis modules update this resource by adding one or more transcription hypotheses and mentioning on a dedicated label each moment where the sentence can be considered self-sufficient. The module in charge of detecting the end of the utterance also helps itself in real time to this *utterance* resource, and indicates that the system can start talking when the prosodic and syntactic parameters allow it. The semantic and pragmatic modules then start to process the utterance, to enrich it and if the dialogue manager judges it relevant, it might decide to start talking, with the potential consequence of interrupting the user if, while the message was generated, he is still talking (and in that case, speech recognition and prosodic and syntactic analyzers keep on working). Systems, such as NAILON (Edlund *et al.*, 2005), open up the way for this type of operation, at least for prosodic aspects. The interesting factor is the *utterance* resource, which is constantly evolving as the analyzes are carried out: far from being static data, it becomes dynamic data, sometimes vague or incomplete, which system modules will use as they can without any strict completion or correction (e.g. grammatical) constraints that are too often imposed. As we can see, the matching run-time architecture requires serious reflection on the resources and modules involved in the acquisition of knowledge and the update of the data.

4.1.2. *The process flow*

The flow of processes carried out by the different architecture modules can happen in a linear manner, as we have seen, that is as a chain: the output of a module directly serves as the input for the following module. From a computational point of view, this is a constraint that can be the source of the implementation of run-time architecture as it has been defined. It is the difference between conceptual architecture and its

materialization as software architecture: a chain software architecture only makes sense if the conceptual architecture itself is a chain architecture.

One type of software architecture used at the beginnings of HMD is the *blackboard*: the knowledge is brought together in a kind of database that is accessible to all the other modules, a bit like we have just illustrated it for real-time analyses. Each module can be dependent on restrictions covering components in the database, and may only see a part of them. In that case, the software architecture has a supervisor that determines the knowledge to activate at every moment. Compared to chain implementation, blackboard provides more flexibility and allows various modules to collaborate. Obviously, it is necessary for the processes generated by the modules to allow this type of collaboration: if the modules are implemented in the same way as the chain model, these processes can only happen in a chain, whether the data they require are accessible in the blackboard or whether it travels from module to module. This is another difference between conceptual architecture and software architecture: the blackboard is useful so long as the conceptual architecture has planned for modules that share the same data. The GUS system agenda is an example. In modern implementation, we should note that the current blackboard, at least for a connected HMD system, can be... the web (especially with the success of cloud computing).

(Sabah, 1997) suggested an interesting software architecture in the mid-1990s, called a sketchboard. In this implementation, each module is able to assess what it generates while taking into account the inputs it has received. Thus, when the syntactic analysis module receives the transcription of an utterance, it runs a syntactic analysis, generates a sketch and calculates a satisfaction score in relation to this sketch. This satisfaction score is transmitted to the other modules, more specifically to the module that generated the data used in input. Depending on the score, it may restart its work and generate a new transcription, transmit it and thus encourage the syntactic module to generate a new, hopefully better, sketch. This is in fact an extension of the blackboard with feedback. For the modules not to generate the same analyses every time, noise is introduced in each stage. Other methods can also be considered to replace this noise that may lack relevance: taking out a setting used by the user or modifying the importance of a setting, for example, such as prosody.

The most successful software architecture is probably the multi-agent system. Each module is matched with an agent in charge of its interactions with the rest of the architecture. Thus, this agent manages all the inputs and outputs, and can also carry out similar processes to those in the sketchboard. With such a materialization, any problem is solved by the converging interaction of different agents. The TRIPS system, a successor of the TRAINS system (Allen *et al.*, 1995), is implemented as a multi-agent system, with approximately the following agents: interpretation, dialogue management, generation, discourse context, referential context, task, which all communicate with each other. The exchange possibilities are thus very numerous, and it

is the interaction between agents that allows the dynamic data to stabilize itself and create a satisfying result.

However, the need to manage the dialogue along various levels or priority, especially – if we follow our example with the *utterance* resource – at a real-time level very close to the speech turn management and a more reflective level matching dialogue strategies, has led to the rise in N-tiers architectures, with new constraints and new system specifications. (Rosset, 2008, p. 83) mentions a set of approaches of this type, with the typical example of double-layer architecture to manage in a simultaneous and asynchronous manner the short-term behaviors, such as starting to talk, and the long term behaviors, such as task and dialogue planning, but also triple-layer architectures, able to manage various aspects of communication separately, such as with an ECA. Similar multilayer architecture have also been used for ages in the HMI field, with interaction management that separates various logics: the persistence logic that concerns lasting data, the application logic that concerns task management, the interaction logic that concerns user action management and the data presentation logic that concerns real-time display, so as to not present obsolete data or data that has just been acted on by the user. By rationalizing the design, these architectures allow us to adapt them to different terminals and different work contexts. The generalization of this approach to architecture for interaction system in a broader sense is recent. It allows us to integrate software design models and human-machine communication models. Thus, (Lard *et al.*, 2007) offer a hybrid approach meant to specify architecture for systems that bring HMI and HMD together.

4.1.3. *Module interaction language*

Whatever the chosen architecture, data are exchanged. A format is thus required to proxy the data in a standardized manner, so that each architecture module can decode and use what it might need. As for architectures, there might be as many interaction language proposals as there are systems, especially in multimodal dialogue, due to the diversity in possible modalities and the type of content that is used. (Denis, 2008, p. 93) mentions a few of them and highlights the MMIL language, which was used in the MIAMM, OZONE, AMIGO or even in the MEDIA assessment campaign (Devillers *et al.*, 2004), in which he participated. This language takes the shape of a standardized file, which allows for a representation of the communicating events in an HMD system, whether we are talking of multimodal external events (speech and gesture) or internal events (exchanges between modules). The point is to represent both the events and the content of the messages exchanged. The representation also takes place at various levels, and it includes, for example, a representation of the semantic content of an utterance or even its speech act. The content also remains as close as possible to the utterance without much interpretation. The semantic content is thus very close to the utterance's structure, and does not integrate any representation of the implicit or any

formalization in a logical form: it is the choice that has been made, the enrichments are managed by the modules in charge and not at the global architectural level.

One of the aspects of MMIL has also been the focus of various works linked to run-time architecture materialization. They are the progressive overlapping of human-machine interaction and module interaction, which is purely computational. From the point when the human-machine communication is formalized as commands and queries covering semantic content, there is nothing to prevent us from doing the same for communication between two modules in an architecture: a module that needs information to finish an analysis may query another module just like a module that has just finished an analysis may communicate it as an assertion. Even if this is not an essential challenge, since the complexity of human-machine communication is much greater, it is a way of applying pragmatics research to the design of a consistent interaction language.

4.2. Design-time architectures

Generic HMD system research, that is partly set and reusable, can go through the specification of a generic run-time architecture. In this case, a specific amount of attention is given to determine the modules and sub-modules, the interaction language and the task's setting procedures. The latter is specific to the system, and all that concerns the task is managed in an independent manner. The task objects are grouped in a database that becomes one of the domain model's settings, the task's specific lexicon becomes a setting of the lexical module, and so on. If the task allows utterances with a specific syntax, with, for example, the disappearance of certain prepositions and certain articles, as it can be seen in specialized languages, then these specific syntactic structures also become one of the parameters of the syntactic module. We then separate all that is task specific (settings) from all that is common to the whole system (knowledge and processes).

Another way of solving the issue of genericity is to put efforts into designing an HMD system development environment. A system is not run autonomously, but becomes the product of a kind of system factory. This development environment, or toolkit, will allow us to import the task's specificities and to start the creation of the targeted system on this basis. We then enter into the field of design-time architecture, that is the design of the development environment rather than the HMD system itself.

4.2.1. *Toolkits*

Toolkits provide system developers with programming interfaces, i.e. the technical abilities to implement the techniques used in HMD. There are other kinds of toolkits that often integrate an architecture model as well as offer techniques. They also offer

a platform to support the developed code's running. More evolved, they can also emit recommendations based on the model they implement, in relation with the software development standards.

(McTear, 2004) highlights the use of toolkits. In general, most of the major HMD research laboratories and computer science companies offer their own development environment based on their experience in running systems. Among other examples, we can mention the famous VoiceXML, the CSLU toolkit or that of Carnegie Mellon, which are regularly taught in tutorials or summer schools. The observation which is often made is that the available toolkits help develop simple systems, but are far from allowing a programmer to design a natural dialogue system in natural language. To achieve this goal, all the NLP aspects have to be implemented in various languages, and that falls more within the scope of NLP and HMD challenges. However, the tools offered help develop systems and allow programmers to start designing an HMD system without having to start from scratch as was the case in paragraphs 3.1.1 and 3.1.2.

More recently, efforts undertaken in the field of model-driven engineering (MDE) and in its subset that concerns architectures, i.e. model-driven architectures (MDA), have reached the attention of human-machine communication researchers, starting with the HMI specialists. These have started reflections on the rebooting of the design cycle of an HMI so that models and metamodels are taken into account during the very first design phases, the last stages consisting of the automatic generation of HMI by the development environment. Such reflections started off in HMD, and pursuing them is a major challenge for the field. More specifically, the designer specifies design models and model transformation processes (in a top-down direction between the models). The toolkit then generates models, and, after one or more generation and deriving phases, ends up by generating code that can be run. Today, these highly software-oriented approaches do not sufficiently take into account the needs of the users and cannot be easily adapted to processes that are as complex as those of automatic natural language understanding. We have to admit that the definition of models and metamodels is problematic, especially if we want to take into account the specificities of natural language and natural dialogue, let alone multimodal dialogue: a common representation language for all the models has to be defined, so as to help the development environment manage them. Yet acoustic, lexical, syntactic, semantic, pragmatic models, models describing the structure of a dialogue, or even models for pointing gestures, involve a huge variety of knowledge for which there are no interconnected representation standards. Moreover, processes for speech recognition, syntactic analysis, reference resolution, mental state attribution, etc. are also the focus of representations in dedicated models. And if these descriptions and formalisms exist, there is still an important effort to be made to achieve usable representations in the case of a model-driven approach.

4.2.2. *Middleware for human-machine interaction*

An alternative, which may join the *OpenDial* challenges described at the end of Chapter 1, is in the design of middleware adapted to the HMD's need. Middleware is an interconnection software component that consists of a set of services allowing multiple processes to run on one or more machines and interact through a network. Closer to our concerns, middleware is also, and above all, a software component that becomes a conversion and translation layer between two processes. Originally, middleware was essentially a system middleware, that is a conversion layer inserted between a flow of data generated by a specific machine and the machine's operating system. We then distinguish explicit middleware (proxy layers between the operating system and any application run on it) and implicit middleware (mediation or interpretation process between a business application and the matching presentation application). The field of HMI has led to the development of various pieces of implicit middleware. (Lard *et al.*, 2007) offer an implicit middleware based on a multilayer architecture to promote the development of human-machine interaction systems by including a few very simplified HMD aspects. The principle resides in adding a layer devoted to the human-machine interaction in the original multilayer architecture. This layer is implemented as *interaction middleware* and provides generic services for human-machine interaction, as well as an abstraction of application specificities and uses of contexts. This proposal focuses on the technical aspects with regards to the general architecture and practically does not mention the NLP issues and natural dialogue in natural language issues. However, this is a path that is a challenge to help the HMD system development process.

4.2.3. *Challenges*

As we have seen, the paths taken toward running and publishing design-time architectures are still in their infancy. They allow us to draw up a list of the challenges for the years to come more than to draw a conclusion on the advances that have actually taken place. Indeed, the design of many current systems bypasses the design-time aspect, the specification of a run-time architecture rallying most of the efforts.

In the previous sections, we saw that one of the first stakes was the establishment and broadcasting of the development environments that were best adapted to HMD, that is to the NLP aspects in all their complexity, right up to natural dialogue management techniques. A second challenge consisted of applying the model-driven engineering approach to the HMD, taking into account once more the diversity of aspects involved, and thus implementing a large variety of models, which tend to echo the list mentioned in paragraph 3.2.5, by adding all the technical aspects, and notably the system adaptation aspects (plastic user interfaces). A third challenge is the specification of an interaction middleware that has to be more elaborate than the middleware presented in (Lard *et al.*, 2007), with once again taking into account NLP aspects in all

their variety and complexity. This challenge can actually be multiplied if we consider that such a piece of middleware would be useful for each of the domains of NLP, for ECA or even for AI with a set of services devoted to different types of logic reasoning.

Finally, one challenge that is often put forward in AI is the design of systems able to assess and modify their own algorithms. With the derivation principles of models and model-driven architectures, this old dream might become reality. Indeed, an HMD system obtained by deriving models may, during its running, update certain models on which it relies. If it has learning abilities, it might want to enrich a model with new knowledge. If it has confidence or relevance calculation scores, it might want to set the data of a model by using these scores, so as to reinforce the impact of certain settings compared to others. Theoretically, as the HMD system is increasingly used, so it is susceptible to question the models on which it has been built. We could then imagine a phase in which the HMD system decides to update itself on its own by restarting all the derivation process.

4.3. Conclusion

A finalized dialogue system is designed to help the user carry out a given task. Any system is thus optimized for a specific task and reveals itself to be poorly performing for a different task. If we take into account the major effort in design, the question arises of reusing the components from one system to the next, and beyond, the question of the possibility of designing generic components, which can be used regardless of the task since they would have flexible settings. This chapter describes the question of genericness through design and software architecture examples, with, on the one hand, the notion of run-time architecture, to the system's running, and, on the other hand, the notion of design-time architecture, with, for example, the model-driven development, whose principle is to go from a set of models describing resources and processes and automatically derive the modules that will constitute the final system.

SECOND PART

Inputs Processing

Chapter 5

Semantic analyses and representations

This second part is focused on the processing of the user's utterances at the system's input, and it starts with what we can consider to be the core part of a system's understanding abilities: determining the meaning of the utterances. The goal is to achieve a computational representation of this meaning from the input audio signal and the system's internal context representation. The aim is to achieve a representation that does not involve pragmatic aspects such as reference resolution or the determination of the speech act (see Chapters 6 and 7). Calling upon context is, however, crucial because in a dialogue an utterance can follow up on the previous utterance, for example by adding an adverbial that can only be interpreted thanks to the previous utterance, or an utterance can take up a term that had been the focus of a lexical explanation, or even an utterance can be elliptic, that is omit a term, such as a verb or a noun, due to the previous use of this constitutive element. Determining the whole meaning of "I want the shortest one" after talking of a journey to Paris can thus integrate the concept of journey, in the same way as "and how long with the other?" after "how long will this journey take?" can integrate "take" or at least integrate the fact that the question focuses on a lapse of time and not a distance. In the example given in the introduction, if we consider that it is an oral dialogue, we can also consider that, due to context, the utterance "how long with this itinerary which seems shorter?" does not match the noun "seam" like in "how long with this itinerary which seams…" (in French, with the same example "combien de temps par ce chemin qui semble être le plus court ?" there can similarly be a confusion between "par ce chemin" and "part ce chemin"). There are many various examples, and this is what makes this process difficult. Here, our approach is not to suggest a model of meaning determination (there are dozens already), but to give an idea of the linguistic phenomena that appear in HMD and of the understanding processes to consider in order to deal with them.

The first section of this chapter thus aims to describe a few phenomena coming into play in oral and multimodal dialogue (section 5.1), and the remaining two sections give an idea of the computational processes involved in determining meaning, with a set of processes that analyze the utterance's characteristics (section 5.2) and a set of processes that enrich them with supplementary elements to reach a representation which can be used for dialogue management (section 5.3).

5.1. Language in dialogue and in human-machine dialogue

5.1.1. *The main characteristics of natural language*

An utterance is made up of words, each of which has one or more meanings. In a given situation, some meanings are impossible and we try to identify the meaning taken by the word in context. Some words come together to become a constituent characterized by its function in the sentence (subject and direct object) and by its thematic role (agent and patient). Some of the constituents refer to specific objects of the context. Even if the meaning and reference are linked, this specific point is the focus of Chapter 6. Like the other full words, the verb of the sentence has a meaning, and it is first and foremost this meaning which will allow us to link all the constituents of the sentence together and to be able to determine the meaning of the utterance. In particular, with our example of trains, but in general, the meaning can involve the notions of time and space, and in that case the understanding of the utterance involves knowledge of temporality (notion of date, finding timetables, length of a trip and placing an event with regard to another) and the semantics of space (notion of place and notion of journey). The characteristics of natural language are thus multiple, and we will focus on the following aspects, each time showing a few examples of how HMD systems should take them into account: polysemy, metonymy, metaphor, verbal semantics, implicitness, ambiguity and information structure.

The confusion between "seems" and "seams", which can happen in an oral dialogue shows us an example of two homophones. Homophones are words that are pronounced in the same way, whereas they have different meanings. Furthermore, language has the particularity that a single word can have various meanings, a phenomenon called polysemy. Thus, all the words of language, or almost all of them, are polysemic. The train ticket that we have mentioned earlier can refer to the piece of paper or the right to a seat which it represents, and if we just limit ourselves to the word "ticket", without taking into account the compound noun "train ticket" the number of possible meanings has gone beyond 10. Polysemy comes from metonymical or metaphorical relations (see later), from meaning restrictions, extensions ("minute" refers to a very specific amount of time, but also, by extension, to a very short period of time) or even phenomena linked to the differentiation of a feature that leads us to take two notions into account. For NLP and open domain HMD, there is a challenge in describing these transformations (or using a parallel path based on the statistics on

joint occurrences) so as to automatically deduce the possible polysemic meanings of a starting point. For a closed domain HMD, we will have to suffice with multiplying the lexicon entries and keeping the most relevant meanings when taking the task into account.

Language also has the particularity that one word can take the place of another word. Metonymy thus takes place when a container takes the place of the content ("train reservation" for a train seat), a part takes the place of a whole (synecdoche), a quality takes the place of a person ("first class is very noisy"), an instrument takes the place of an agent, an action takes the place of an agent, of the place it is carried out in, of its own effect, etc. For HMD, we will find the two previous techniques: either the lexicon entries and potential sentence structures are multiplied, which leads to a multiplication of possibilities which is manageable when it comes to a specific task, or adding rules allowing for metonymy replacements which enable a fine semantic analysis.

Metaphor is a close phenomenon that consists of giving one word another's meaning, by analogy. Thus the buttons and menus of HMI are called metaphors. If we keep with the train ticket reservation dialogue, the following sentence, "I don't want to ride a snail, I will take the train going through Meudon", is a hard sentence for the system to interpret if, due to the nature of the task, "snail" is not a word in its lexicon. In that case, one of the challenges for NLP or open domain HMD is to find the feature of the "snail" that will allow the system to understand its relation to a train, and thus understand that the user does not want a slow train. In the case of a closed domain HMD, a fallback strategy would simply be to ignore the incomprehensible metaphor and only retain the second part of the utterance to determine the system's reaction.

The phenomena mentioned until now happened at the word level. Yet, some words such as "reservation" have an additional characteristic: they allow us to create links with other parts of the utterance. A reservation is made by someone, the agent, and for something, the object. The identification of these two actants is necessary to interpret the sentence correctly. Thus "reservation" has a valency of two, just like the verb "to reserve" does. The verbal semantics thus rely on this notion of valency, which allows the system to give a meaning to the prepositions used in the utterance and link the different components. In "I am reserving a ticket", the semantics of the verb "to reserve" thus allows us to determine the agent, "I", who is also the user, and the object, a ticket. It is also the case for "I am cancelling my return reservation", both for the verb "to cancel", which has a valency of two, and for the noun "reservation", with the possessive form allowing the system to identify the agent, and the complement "return" allows the system to identify the object. In HMD, the verb identification and its semantics can thus be an entry point for automatic understanding. Taking valency into account does not, however, allow the system to link all the components in the sentence. Thus, in "I am definitely cancelling my return reservation", and "I am cancelling my return reservation because I am staying in Paris", an unnecessary

verb complement is added to give (relevant or irrelevant) indications on the process application conditions provided by the verb. For NLP as for HMD, we have to catalog the different verbs with their characteristics, and, while analyzing them, identify the actants expressed as well as those that are implicit and are thus subject of an ellipsis or a particular use. There are available dictionaries for the first point, and interpretation strategies have to be devised for the second.

The fact that some parameters cannot be expressed is at the heart of an essential phenomenon in language: implicitness. Everything that is not expressed but that is still carried by the utterance falls within the realm of implicitness, which thus covers varied phenomena. Among these we can find ellipses, allusions ("I do not want a snail" can imply "please stop suggesting trains that are too slow"), presuppositions and other inferences that can be found through complex pragmatic analysis and not semantic analysis. Identifying the implicitness is a challenge for NLP and HMD. Beyond the cases of ellipsis that can be modeled by exploiting the dialogue history, implicitness relies on the hypotheses that are hard to formulate. In the case of a specific task, the inferences are easier to observe than in open domain because they preferentially orient themselves toward resolving the task. In open domain, approaches such as relevance theory (Sperber and Wilson, 1995) provide us with an interesting framework that is hard to implement.

With or without implicitness, a natural language utterance is often characterized by the phenomena of ambiguity, that is the possibility of obtaining various alternative interpretations. This multiplicity can come from a term or reference that gives the system a choice between several possibilities even after taking into account the context, such as "I want to leave at six", which could be 6 am or 6 pm. It can also come from the sentence structure which due to the ambiguity of a preposition does not allow the system to link the right components to a noun or a verb. In "I will take a ticket from Meudon to Paris", does the user mean he needs a Meudon–Paris single ticket, or that he will take the ticket from Meudon and bring it to Paris? The challenge for NLP and HMD is to identify the terms and structures which might generate ambiguities, based, for example, on a grammar-type inventory (Fuchs, 2000), that is a set of rules on specific words and sentence word order.

As we shall see in section 5.2, this kind of set of rules can also be used for sentence syntactic analysis, at the same time or separately from semantic analysis. The elements mentioned above allow the system to have an idea of the extent and complexity of processes. One last aspect on which we would like to emphasize here is the *information structure*. This notion describes the fact that some actants are highlighted compared to others according to the mechanisms that go from syntactic structure, including word order and specific construction use, to prosody. We can, for example, distinguish between previously known information and information given by an utterance, between the topic (that of which the utterance is speaking) and the commentary (what is said about it), or even between the focalized (the focus, which is given a

prosodic prominence) and that which is not. These dichotomies lead to ranking the utterance's components, and this ranking becomes a point of view in relation to the semantic content. Thus we can draw a line between "I want *a* first class ticket for Paris", "I want a first class ticket *for Paris*" and "I want a *first class* ticket for Paris" (the italics indicate the accent), after, for example, the same utterance without an accent was given to the system which made a mistake. Moreover, the information structure also makes sense beyond the boundaries of the utterance, and we can thus add a topical progression characterization to semantic analyses, which describes how topics follow one another in a dialogue and what relations are built between one another. In HMD, information structure identification and topical progression allow us to better manage the way in which the current utterance can be inserted into the dialogue history, and thus better manage the dialogue and achieve satisfying the task in a more efficient manner, for example by coming back on the main topic if the dialogue has deviated too far.

5.1.2. *Oral and written languages*

Oral and written languages are not only different due to their means of expression, to prosody with the prominence accent we have just mentioned, but also by their use of syntax and morphology: there are many different graphemes and phonemes, different marks of the plural, as well as the importance in enunciating the links ("liaisons" in French) between words[1], see paragraph 2.2.1. Different language levels can coexist in oral language. Utterances such as "could I get a train for Paris?" can alternate with "a train for Paris, is that possible?" which at the end provides the system with a multitude of possibilities (Cohen *et al.*, 2004). The oral language ignores some *errors* more easily than the written language such as tense agreement or other rules with verbs. The oral language is characterized by noise (hesitation and interjection), distortion (repetition, specification with no cancellation and correction with cancellation), fragmentation (starting again after an interruption, juxtaposition, dislocation and sentence fragment) and ellipsis phenomena which can appear in written language, but much less frequently, and lead to diversified syntactic structures: "what time for the Paris train?", "at night, the Grenoble train, where does it stop?", "is it the shortest journey, the one going through Meudon?" or even "I'd like two of them, those tickets". The consequences for HMD are the increase of potential structures that the system has to know in order to process the input sentences.

Oral language greatly emphasizes *macrosyntax* (Blanche-Benveniste, 2010), that is a grammatical organization which, contrary to syntax, is not based on grammatical

1. In French, consonants which would normally be silent at the end of a word are sometimes vocalized when this word is followed by another word starting with a vowel. Thus, while the final "s" is not pronounced in "moins" (less), it will turn into a "z" sound when placed in front of "agréable" (enjoyable).

categories: the units are utterances, with components such as the core (at the center), the prefix (which is placed in front of the core and partly matches the topic), the suffix (which is placed after the core) and potentially the postfix (which is placed after a suffix). Macrosyntax notably allows us, and this is what it is interesting in HMD, to chart links between successive constructions which are neither coordinated nor subordinated as they might be in written language, but that constitute a whole, thanks to the prosodic criteria of intonation period. The latter matches a segmentation of the utterance on a set of criteria which include the distribution of pauses and variations of pitch (melody). Software such as ANALOR (Analysis of Oral) knows how to detect them automatically, but in a corpus and not in real time. The automatic use of macrosyntax in HMD is still a challenge.

Finally, oral language is obviously the field of prosody, which covers accentuations, i.e. the highlighting of a unit compared to others (as we saw with prominence), the rhythm (allocation of pauses, speech rate and its variation in a single utterance or between various utterances) and intonation, a melody curve that allows us to give an illocutionary value to an utterance (order, question and assertion). (Rossi, 1999) offers a description of prosody in French language, and some aspects of it can be applied to HMD. The challenge consists of implementing a real-time analysis able to detect rhythmic groups which constitute units, of detecting accentuations, on more or less correlated intensity and duration criteria, which focus on units or on the utterance as a whole; to identify intonation periods to help with macrosyntactic analysis; and to describe the melody curve of rhythmic groups, to annotate them with potential illocutionary values and types of modality, a modality being a modification of the content uttered by presenting it as necessary, possible, probable, etc. (paragraph 5.3.1).

5.1.3. *Language and spontaneous dialogue*

In a dialogue, since the speaker and the hearer are talking to each other, terms of address can occur: "*you* show me the trains for Paris", "hey, *machine*, you're being slow!". These terms can be used to refer to the hearer, first and foremost to attract his attention, and do not create any specific issues in HMD, especially when there is no ambiguity in the identity of the hearer, which would not be the case in a polylogue. We have to just plan a process so that either they are ignored in the syntactic structure of the utterance or a positive or negative qualifier is taken from them and used to react accordingly.

In general, the way in which the user addresses the machine can take on various forms, due to the lack of social status that the machine has: "give me a train ticket for Paris" (imperative, easier to use with a machine than a human hearer), "reserving a single ticket for Paris" (gerundive form which is specific to HMD, in French it would be the infinitive form, "réserver un aller pour Paris"), "ticket for Paris please" (elliptic form), "we are going to reserve a single ticket for Paris", "and then I get a ticket for

Paris", etc. Moreover, the utterances are not always complete sentences – see Chapter 10 of (Cohen *et al.*, 2004) – which causes various issues in HMD. On the one hand, there are issues for the detection of the end of the utterance, since only the prosodic criterion may help the system decide. On the other hand, there are issues for the syntactic analysis: if it produces anything, the result is an incomplete structure, which might be false when the analyzer is not adapted. Adaptation is also necessary, as (Vilnat, 2005) shows us, for example. Finally, an utterance can be so incomplete that it can only be interpreted in relation to the previous utterance. These are the answers to questions, these are also all the *non-sentential utterances* (NSU), such as "Ok", "thank you" and "sorry", which are notably studied by (Ginzburg, 2012).

5.1.4. *Language and conversational gestures*

In a human dialogue carried out face to face, gestures play an important role both in the organization of the interaction, in the transmission of emotions or modalities and in the designation of various elements of the shared visual scene (Kendon, 2004).

Language and gesture supplement each other in spontaneous communication (Landragin, 2006), which leads us to favor, when technical means allow us to, the multimodal HMD over the oral HMD. The gestures thus allow the user to speak, keep his speech turn and let the machine know that he understands what the system is telling him. These synchronizing gestures can be interpreted without the need for an utterance to be pronounced at the same time. In HMD, a recording device like a camera and shape recognition algorithms are required. Gestures are also what transmits a major part of the emotional component of the message: expressive expressions, such as facial expressions, and paraverbal gestures, such as the movements that give speech rhythm and emphasize certain words. There are also the gestures which allow the user to provide information on the referents in the utterance, whether it is by pointing (pointing or deictic gesture) toward the referent in question, when it is visible in the shared visual scene, or an illustrating gesture that indicates a size, shape or action. Finally, there are also gestures which can carry a dialogue act such as a questioning gesture (wide eyes and raised eyebrows) or a quotation gesture close to the reported speech, like pointing to the hearer to mean "as you said earlier" (Clark, 1996, p. 258).

In the same way, and even if a gesture is not necessarily present, the dialogue language can have "deictic" terms, that is the terms or expressions that refer to the communication situation, with the three categories that are person deictics, which refer to the hearers thanks to markers covering personal pronouns, possessive adjectives and some nouns; spatial deictics such as "here" or "there" and temporal deictics such as "now", "tomorrow", "later" or "in three days". For deixis cases, an HMD system has to find a link between the deictics pronounced and the situational context. This link relies on the hearers, on the objects in the scene and on the spatial and temporal

markers which will serve as parameters in the semantic and pragmatic representation of the utterance.

5.2. Computational processes: from the signal to the meaning

The automatic understanding of an utterance starts with the voice recognition and prosodic analyses, and ends with the enrichment of a semantic representation through inferences carried out essentially from the utterance's content. This representation will then be analyzed with a pragmatic approach that allows us to use it in dialogue management. The processes involved in building this semantic representation can be deduced from the content of the previous section: lexical analysis, detection of polysemy, metonymy and metaphor. While we will not draw that list again, this section aims to present the general processes by mentioning methodological and technical solutions and underlining the most crucial challenges.

5.2.1. *Syntactic analyses*

Syntactic analysis consists of highlighting and representing the structure of a sentence in computational data, with its constituents – the verb, the subject, the direct object complement, etc. To this end, it requires a list of language words with, in each entry, the category (verb and noun) and morphological properties (gender, number and person). The latter helps it manage the morphology at the same time as the syntactic analysis is running. Thanks to the work of R. Montague (Muskens, 1996), automatic understanding, especially for written dialogue, took the path that consists of carrying out a syntactic analysis of the sentence before a semantic analysis, which will itself lead to the determination of the logical form or forms describing the possible meanings. A pragmatic analysis that takes into account the contextual aspects then enriches this semantic representation to achieve a propositional form, the result of all the automatic understanding processes.

To process spontaneous oral dialogue, this process, which goes through a global syntactic analysis of the sentence, is not always adapted: as we have seen earlier, a dialogue utterance can be an incomplete sentence, and other mechanisms have to be implemented for the system not to crash due to a syntactic analysis which is impossible to run. We then have to implement syntactic analyzers able to create partial analyses, i.e. able to manage the underspecification of an actant, for example, or local analyses, which generate a result with the available data even if it is incomplete.

Whether it is a global or a local or partial analysis, or even a macrosyntactic analysis, they all supplement each other so as to use the input data as best they can. The local analysis principle notably allows us to temper the importance of syntax in the understanding process and to highlight the semantic analysis, which, in a way, leads

to the operations and requires local syntactic analyses when necessary. This type of mechanism is more adapted to the characteristics of oral language, such as distortion or fragmentation phenomena, or simply the frequent presence of questions, a type of sentence that does not appear very often in written texts, and is thus not well taken into account by analyzers which are designed and trained on written texts. It can also turn out to be more robust to potential errors in speech recognition. This is especially the case in closed domain HMD: for a train reservation task, we almost know in advance all the elements that a user query can cover, and these elements can thus help the analysis according to a top-down approach and supplement the bottom-up approach started with speech recognition.

Thus there are a multitude of ways to implement a syntactic analysis, just like there are a multitude of representation formalisms. Depending on the user's utterance, an implementation can turn out to be performing more than another at certain moments and for certain phenomena. If we have the required computational means, we can consider, as (Vilnat, 2005) did, implementing several algorithms, i.e. running a *syntactic multi-analysis*. The idea is to extract a trustworthy analysis from the results of various analyzers, each result can be accompanied by a satisfaction score, just like in the sketchboard approach seen in paragraph 4.1.2. The observation behind this approach is linked to the success of processes which combine multiple outputs, both at the level of combining hypotheses made by the speech recognition module or combining morphosyntactic labelling hypotheses (Vilnat, 2005, p. 20). If satisfaction scores cannot be calculated, one strategy is to favor the results that have been produced the greatest number of times by all the analyzers involved. In that case, it is useful to maximize the number of analyzers, knowing that the real-time HMD constraints might slow down this approach: even if the syntactic analysis is not the most greedy in terms of memory and calculation time, it is good to be prepared.

5.2.2. *Semantic and conceptual resources*

The list of words in a language with their category and morphological properties is the main resource necessary for syntactic analysis. Other resources can also be used, such as statistical data on the word succession and possible sentence structures. To go further and discuss the meaning, additional data are required. It is first about the meaning of words, or at least of a formal representation, for example a feature structure that puts aspects into boxes, such as object types (concrete inanimate, abstract, animate and human individuals), types of properties (gradable or non-gradable), events (actions and processes, which are limited in time) and states (not limited in time). The language dictionaries, unfortunately, cannot be used to fill such structures for they are not structured enough to express the meanings in natural language, which makes it a circular problem: we have to automatically interpret the definitions of a dictionary to obtain useful resources for automatic interpretation.

Various approaches are used, if possible in a complementary way, to build what is called a *semantic lexicon*. The component approach aims to specify, for each word, the set of semantic features which describe the meaning of the word and thus allow the creation of a lexicon, which is relatively close to the way a dictionary works. In HMD, this is a completely realistic approach from the moment when we place ourselves in a closed domain, that is with a limited list of words. The lexical database approach like in WordNet allows us to obtain an organized list of words, without necessarily equivalent definitions (a thesaurus is drawn up rather than a dictionary), but with many oriented links between the words: hyponomy ("is a kind of", e.g. from "TGV" to "train"), metonymy ("is a part of", e.g. from "coach" to "train"), troponymy ("X is Y in a way", one of the kinds of implications between two verbs), antonymy (opposite), synonymy, etc. These relations, especially since they have a direction, allow us a greater analysis possibility than a purely component approach would: we can thus model the fact that a "long" journey is "not short", but that a "not long" journey is not necessarily "short". Finally, the approach of derived lexicon, built from corpus or resources such as those obtained through the previous approach by following the principles of a specific lexical theory, allow us to obtain richer data structures. A famous example that has often been used in HMD, within the framework of a limited task, is that of conceptual graphs (Sowa, 1984). This is a kind of formalism that allows us to go further in the representation of knowledge, with multiple relations between concepts, and which leads to the notion of *ontology* as a representative data model of a set of concepts in a domain, allowing us to reason on the objects falling within the scope of this domain. An HMD system focusing on the train ticket reservation needs not only language resources, for example a semantic lexicon, but also resources on the world of trains, transportation and reservation, with the matching ontology, whether it is represented by a conceptual graph or a set of feature structures.

Moreover, some syntactic formalisms use lexical data such as that constituting a semantic lexicon, which leads to implementing lexicalized grammars (Abeillé, 2007), and thus mix lexicon, syntax and lexical semantics. One of the HMD challenges is to create and use these semantic syntactic lexicons, especially in an open domain HMD. We have mentioned earlier the WordNet initiative, and can mention here, as one of the paths that is currently being investigated by linguistics and NLP in general, the FrameNet initiative that consists of cataloging sentence constructions, by linking them to each other as soon as they describe a similar meaning. It is a kind of semantic syntactic lexicon leading toward the automatic identification of thematic roles, but not exclusively, and this is the type of initiative that HMD can use.

5.2.3. *Semantic analyses*

Semantically analyzing an utterance first means determining the meaning of each full word used, which a semantic lexicon allows us to do, and also means building the sentence's meaning. This can be done from the structure identified by the syntactic

analysis: following the relationships between components allows us to build semantic relationships. The analysis thus uses a semantic syntactic lexicon and is based, for example, on the semantic features and grammatical functions to determine the thematic roles of the different sentence components. This is one of the key roles of a semantic analysis, with notably the works of C. Fillmore highlighting this aspect in case grammar, following the observation that the syntactic structure is not enough to explain the links between a verb and its actants (in general, see Enjalbert, 2005). Another role is to choose among the possible meanings of a word by confronting the semantic features of the elements present, i.e. solving the polysemy cases and other aspects of language mentioned in paragraph 5.1.1.

For a closed domain HMD the semantic analysis can almost stop there, because the determination of verbal semantics and the identification of the actants allow us to guess the semantic content of the utterance, at least in the simpler cases such as "I would like to go to Paris" or "I would like to reserve a single ticket". For an open domain HMD or for systems which try to achieve a fine understanding of linguistic phenomena, another role of the semantic analysis is to express the meaning in a logical manner so that inference calculations can be run on this logical form and on those that have already been recorded as the dialogue progressed, and thus model a part of the implicitness. One method consists of following the mathematical logic principles and to carry out a calculation of predicates, i.e. the formalization of the utterance's content with variables, relations, predicates, logical connectors (conjunctions, disjunctions, implications) and quantifiers (universal, such as "all the trains have first class coaches", or existential, such as "a train has broken down near the Palaiseau station"). The challenge is then to formalize the natural language with the constraints of logics, which creates an incredible amount of questions and issues. This is how modal, temporal and hybrid logics can be applied to HMD.

Beyond the boundaries of the utterance, it is on this path that theories such as the File Change Semantics (FCS) and the Discourse Representation Theory (DRT) will suggest formal frameworks for utterance interpretation (Kadmon, 2001). DRT, which describes how discourse representation structures are built (Kamp and Reyle, 1993), is especially the focus of many extensions or adaptations for computational implementations, for example Discourse Representation Language (DRL), see (Kadmon, 2001). For HMD, an important extension is Segmented DRT (SDRT) (Asher and Lascarides, 2003), which reviews all the aspects of the dialogue, and the implementations that have already been carried out, especially in some QAS, or in the VERBMOBIL project, see (Cole, 1998). Other extensions take into account the underspecification phenomena found in oral language, such as Compositional DRT (CDRT) (Muskens, 1996) or Underspecified DRT (UDRT). Moreover, DRT even warrants an extension for multimodal dialogue, with the integration of linguistic aspects and visual and gesture aspects in the same formal framework. This is Multimodal DRT (MDRT) (Pineda and Garza, 2000).

A completely different path is the use of probabilities so as to obtain a probabilistic semantic grammar. This is what has been done in the TINA project (Seneff, 1995). Other approaches use Hidden Markov Models by adding a ranking structure to combine the advantages of a semantic grammar with those of statistics. (Jurafsky and Martin, 2009, p. 859) thus present Hidden Understanding Models (HUM).

The semantic analyses can thus happen in various ways and various approaches can be combined. Two remarks to close this section: first, as we underlined with the importance of local or partial syntactic analysis compared to a global and comprehensive syntactic analysis, the semantic analysis can also remain incomplete. As (Enjalbert, 2005, p. 303) writes, "we absolutely need to let go of the idea of 'complete' understanding, which is incompatible with the extreme variety of issues to be dealt with. The understanding goals must be brought back to a specific task, which will direct, limit the analysis and provide additional information. Moreover, is it not thus that the human reader works, recording some information in a text, depending on his interests, on the goals for his reading – even if he has to come back later for a more in-depth reading?" Second, determining the meaning of an utterance is not only a semantic issue: it is also a pragmatic issue, obviously, with the determination of referents (the variables in the logical forms) and the identification of the speech acts. The latter allows us to understand elliptic utterances and non-sentence utterances. Pragmatic analysis is thus useful for semantic analysis, and semantic analysis is one of the settings to identify the pragmatic characteristics of an utterance. Semantic analysis and pragmatic analysis thus go hand in hand (Cole, 1998, p. 189).

5.3. Enriching meaning representation

Once we have a representation of the explicit semantic content of the utterance, we can go further in the analysis and integrate a few semantico-pragmatic aspects which fall within the field of connotation, of highlighting or of inferring such as *explicitations* of the relevance theory, that is the hypotheses explicitly communicated through the utterance that indicates, for example, the propositional attitude such as intent or belief (Sperber and Wilson, 1995). We can also enrich the logical form achieved through semantic analysis by integrating a calculation of the *implicitations*, that is the hypotheses that are not communicated explicitly, such as implicit assumptions. These can be derived from the utterance itself, if it is linguistic or multimodal, or from the context. In this section, we are only focusing on those which can be derived only from the utterance.

5.3.1. *At the level of linguistic utterance*

The enrichment of the analyses described in the previous sections consists of adding indications and propositional content to the semantic representation which was

obtained. According to the theories, these indications and this content can vary widely, and for our purposes we will consider the following: connotations, modalities, irony, salience, focus, presuppositions and allusions.

Adding *connotations*, i.e. elements of meaning which are added to the literal meaning, happens by following the links of hyponymy, synynomy, etc., and by identifying the semantic aspects which are linked in a relevant manner to the elements in the utterance, in the cases when such aspects provide useful settings for automatic understanding. For HMD, the challenge is not to automatically generate too many connotations at any random moment, but to generate information that allows us to fill in the blanks in the understanding process, for example to correctly interpret metaphors and comparisons.

The *modalities* are ways of modifying the semantic content by expressing the user's attitude with regard to the content of his utterance. Thus, the term has nothing to do with the multimodality of communication. Depending on the type of modality, we refer to epistemic modality (level of belief in what is said, like in "this itinerary seems shorter"), alethic modality (truth or possibility that what is being said might happen), deontic modality (obligation, ban and permission), intersubjective modality (advice, reproach), or appreciation modality. In HMD, the presence of verbs such as "to seem", "to want" or "to be able to" as well as the presence of adverbs such as "maybe", "possibly" or "apparently" are triggers to automatically identify a modality and use it to enrich the semantic representation.

Irony is the typical example for which the utterance's semantic content does not match the content expressed. It is a phenomenon close to the understatement, and can be hard to detect, even for a human hearer. As always in HMD, it is desirable to be able to detect when the user expresses this type of behavior, so as to bring the dialogue back toward a more neutral path, but we can consider that it is not necessarily a priority, especially in closed domain.

Taking *salience* into account, profiling according to the cognitive linguistic approach (Langacker, 1987, p. 39), allows us to rank the different elements not only in the utterance, for example the number of tickets, the destination and the type of seat in "I want a first-class ticket for Paris" depending on the accents, but also between sentences presented slightly differently, which all lead to the same semantic content: "it is a ticket that I want, a first-class one for Paris", "it is for Paris that I want a first-class ticket", etc. In these utterances, the cleft pronoun "it is" has the consequence of putting some words in salience, and this salience has to be a part of the description of the utterance's meaning. The challenge for the HMD is to implement a model describing all the salience factors, and for each utterance to calculate the saliences linked to each element, so as to rank them and take this ranking into account when determining the system's reaction.

In the same kind of idea, the approaches based on focus, i.e. on salience with a point of view that is mainly prosodic, a point of view that translates the importance of this factor in the English language, exploring the factors which will help identify the focus in an utterance as well as the mechanisms through which the focus effect spreads from a word to a wider linguistic segment (issue of association with the focus). Some approaches, such as the approach of (Beaver and Clark, 2008), consider that a focalized element highlights a set of alternatives which are calculated in a compositional manner as focalized or alternative meaning, i.e. as a set of propositional contents. The challenge for the HMD is to automatically identify this alternative content which adds itself to the semantic content coming from more classical analyses.

To continue on this path consisting of determining additional propositional content, an essential aspect of the interface between semantics and pragmatics consists of the identification of *presuppositions*. This is the implicit content that can be deduced from the utterance's content, as a prior supposition: "hello, I would like to change a reservation for an eight o'clock departure" presupposes that the speaker has already made a reservation, and that this reservation was made for a different departure time. In a less precise manner, we can also suppose that the change only concerns the time, and not the departure station or the arrival. In any case, an HMD system will find it advantageous to identify these implicit contents to be able to react relevantly: to find the previous reservation (and if it needs a name or a number for it, to ask the user for it), to check the settings and potentially ask the user to confirm that the time is the only thing that should be changed.

Finally, the identification of allusions this time relies on a fine interpretation of linguistic hints or on broader contextual information than that which is included in the utterance. In "only the seven o'clock train stops in Valence", one of the potential allusions can thus be through the use of "only", "it could be expected that other trains stop in Valence". Various rules and laws allow us to determine allusions: the law of informativity, which can be applied in this example with "only", the law of exhaustivity, which allows us to infer "some trains do not stop in Valence" from "some trains stop in Valence", the law of understatements, which allows us to infer "none of the trains for Palaiseau are very enjoyable" from "some trains for Palaiseau are not very enjoyable", or even the law of negation, of argumentative inversion, etc. In general, the allusions can cover very different phenomena, as in "I think I will take the plane", which, uttered after a long dialogue in which the user tried to reserve train tickets, can imply "none of your suggestions are acceptable", "you are inefficient", or even "I am fed up, I will stop talking with you". For the HMD, the challenge is to identify the allusions that are closest to the semantic content, so as to take into account the linguistic phenomena such as those caused by the use of "only" or "some". For the HMD in closed domain, the allusions which are less clear can be deduced from the settings linked to the task, for example the difficulty in satisfying the user due to the number of suggestions he has refused.

5.3.2. *At the level of multimodal utterance*

In a multimodal dialogue, the semantic representation of the oral utterance is confronted with other semantic aspects, such as those carried by gestures, expressions and the user's general attitude as it is recorded and identified by the emotion tracking modules and conversational gesture tracking modules.

As an example, a gesture of discouragement, a facial expression close to a frown of distaste or even a set of vocal and gestural hints that translate the user's clear irritation (speed of the movements, speech rate and speech intensity) is a semantic hint which can help clarify the semantic content of the oral utterance. Two main cases present themselves: either the indication given by the other modalities are compatible with oral utterances, for example with the irritation markers compatible with "this will not do, I need a single for Paris and not Lyon", and in this case the indications allow us to specify the mental states of the user (which helps determine the system's reaction), or these indications do not match the oral utterance, and we are probably in the presence of an ironic behavior, an understatement or at least a strong allusion that has to be deciphered and clarified, possibly by asking the user an explicit question.

This confrontation of semantic contents stemming from different modalities is called *multimodal information fusion* and is an aspect that we will recall in Chapter 7 for the confrontation of dialogue acts, and right now in Chapter 6 for reference resolution.

5.4. Conclusion

In a spontaneous oral dialogue, the user's utterance at the system's input is characterized by its prosodic, lexical, syntactic and semantic properties. All these linguistic characteristics create identification and automatic processing issues and lead to implementing devoted techniques depending on the panel of phenomena which has been chosen. This chapter summarizes the input processing and shows how to achieve operational system internal representations. The emphasis is put on the reconstruction of the implicit and explicit meaning of the utterance, so that the system reasons on an enriched semantic representation which is as close as possible to that matching the user's intention.

Chapter 6

Reference resolution

Reference resolution as well as the processes that we will study in Chapters 7 and 8 require input arguments, which can be preprocessed, and give an output result, which might require postprocessing. We are here in the presence of input information, after a pre-interpretation process. We are then in the presence of one or more written transcriptions of the oral utterance (several if there is any ambiguity), with prosodic indications, and simplified representations for gestures. For multimodality to be correctly managed, these transcriptions and representations are accompanied with temporal markers, for example the date stamp of the beginning and end of each word pronounced, of each prosodic accentuation and each gesture trajectory. These aspects are a first set of arguments for reference resolution. A second set brings together the results of lexical, syntactic and semantic analyses as they have been presented in Chapter 5. Finally, a third set of arguments is the dialogue history, in the cases when a linguistic expression refers to a previous reference, which is the case for pronoun and associative metaphors. As is the case for the utterance being processed, the history has two types of representations: representations stemming from linguistic analyses and representations as chain lists of words with temporal and prosodic markers. The first type of representations is crucial to interpret "it" in "put *it* in the box" when it follows "take a green block", and the second type of representations is to interpret "what I thought was a pyramid" or "what I called a strange shape", i.e. for the cases where the referring expression contains properties that do not belong to the referent and cannot be accessed through it.

The result of the reference resolution is an updating of the results of linguistic analyses in which the variables that have remained free are now affected to referents, preferably the right ones, i.e. those which match the user's intention. In case of ambiguity, several alternative representations are created. In case it is impossible

to affect a referent, an underspecified representation is generated and transmitted to the pragmatic analyses modules. Thus, in the example given in the introduction, the U2 utterance has a referring expression "this itinerary which seems shorter", which is accompanied with a pointing gesture and thus refers in a multimodal way to one of the itineraries displayed on the screen. The temporal markers for the oral utterance and the gesture allow the system to check if they are temporally synchronized at the (approximate) moment of the referring act. Prosody can also provide support for this argument, for example if "this" is slightly accentuated due to the simultaneous gesture. The gesture contributes the referent identity, the oral utterance contributes the semantic representation which was made and in which the referring expression has remained as a variable. The fusion process of multimodal information allows the system to solve the reference, i.e. assign the referent identifier to the referring expression.

In this chapter, we will discuss this process: first with object references as is the case for train journeys (section 6.1), then with action references (section 6.2), and finally in the specific case of references which call upon a referent in the dialogue history, with anaphora and coreference phenomena (section 6.3).

6.1. Object reference resolution

Reference has been the focus of many pieces of work, from language philosophy, to logic and linguistics (Abbott, 2010), work which has highlighted the notion of reference, the referring expression categories (demonstrative for "this itinerary"), and a whole set of phenomena characterizing language, as the presentation mode of a referent or the distinction between the attributive use of an expression ("the train for Palaiseau, no matter which one it is, always stops at Villebon") and referential use ("the train for Palaiseau is eight minutes late") that implies a specific referent (Charolles, 2002). In HMD, the issue always comes back to building the link between a linguistic form and an element from a database managed by the application, whether it is a specific train or a type of train, or even the generic class of all the trains as it is defined in the conceptual model. It also happens that the referring expression, for example "a block" in "take a block and put it on the left-hand pyramid", gives the system a choice between several possibilities, the user imposing the referent to belong to a specific and concrete set without choosing among the alternatives: any of them will do.

In general, solving object references is done by using the properties mentioned and the determination of the referring expression. "The green pyramid" thus provides the system with three research criteria in the set of objects available at the time of utterance: the object must be unique and it must be shaped like a pyramid (category) and green (modifier). If the system's database has various objects that verify these shape and color properties, there is an ambiguity. If it finds no object, it cannot solve the

reference. If it has a single object with both properties, then that object is considered to be the referent and the understanding process moves to the next stage.

It may happen, however, that the terms used do not always quite match the way the properties are entered in the database that could explain the times when no referent is found. Going through the conceptual model properties is thus necessary to verify if an object with a similar shape or color exists, an object that might be a relevant candidate. This is the kind of situation that may arise in HMD, since the user does not necessarily have the exact vocabulary that was used to build the conceptual model and the application's object database. This is also the kind of situation that can happen in human dialogue, and that actually shows the whole point of a dialogue: the hearers and speakers can communicate and come to an agreement on the best terms to use given the referent. This convergence toward a common term was studied, in particular, by (Brennan and Clark, 1996), with the notion of lexical alignment: the speaker and hearer align their linguistic behavior, especially in terms of words chosen. Thus, an HMD can do the same thing, which means using the same term as the user, and suggesting another term when he does not understand.

Beyond the properties that appear as full words, reference also involves a variety of formulations, for example the relative clause in "this itinerary which appears in the left of the screen". The challenge for HMD is to understand the meaning of this restrictive relative clause to deduce a research criterion in the object database, in this case a spatial arrangement property that can be calculated with the coordinates of the object as it is displayed on screen. However, in "this itinerary which seems shorter", the relative clause is not restrictive and in fact does not have anything to do with the referent identification, for which "this itinerary" is necessary and sufficient. Finally, let us mention the particular importance of the determiner: with "the green pyramid", the singular definite determiner is translated by a very specific research criterion: find the unique green pyramid in a set of objects that should contain pyramids and objects with other shapes, as well as the green objects and the objects of other colors. The demonstrative determiner in "this itinerary" has a completely different role: it shows that the referent is salient in the communication situation, either because it has just been mentioned, in which case this is an anaphora such as in "the itinerary which goes from Paris to Meudon seems shorter" followed by "I will pick this one", or because it is simultaneously pointed at by a gesture, which is the case in utterance U2. This fine determiner analysis will allow us to implement a performing object reference resolution.

6.1.1. *Multimodal reference domains*

After one of F. Corblin's intuitions (Corblin, 1995) and a set of work carried out in Nancy (Landragin, 2004, p. 107), the notion of reference domain proved its importance for reference resolution in a linguistic as well as multimodal context (Landragin,

2006). The idea is that the identification of referents systematically happens by identifying a contextual subset to which they belong. This subset, which does not extend to the entirety of the context but matches, for example, a focus space, is called a *reference domain*. It allows us to justify the use of the singular definite determiner as in "the green pyramid", even in the cases when the context has more than one pyramid: if there is a previously defined focus space drawn during the dialogue, and this focus space has a single green pyramid, then chances are that the singular definite is not a mistake on the user's part but a localized interpretation in the reference domain, which is this specific focus space.

Compared to the discourse representation theory and its extension for multimodal dialogue MDRT (Pineda and Garza, 2000), the multimodal reference domain model carries out a finer process of the focalization on a contextual subset. Compared to the quantification domain approach (thanks to the work of R. Montague), reference domains last more than one utterance and take into account contextual restriction and broadening mechanisms as the dialogue progresses from a referential point of view. We mention them here for reference resolution, but they are also used for automatic generation of referring expressions (Denis, 2011). They have also been applied to wider phenomena than reference, for example dialogue management (Grisvard, 2000), when linked with mental representation theory, from which it also arose, see Chapter 6 of (Reboul and Moeschler, 1998). Among close works, we can also find those of (Beun and Cremers, 1998) on focus spaces, those of (Wright, 1990) on referential domains or even, for an extension of the same principle to discourse, those of (Luperfoy, 1992) on discourse pegs.

A typical example of how reference domains are used, revolving around the introduction's example, can be highlighted by replacing U4 by "and how long with the other itinerary?": "the other itinerary" can only be correctly interpreted in a reference domain that has two itineraries and one of them is already focalized. As "here are the possible itineraries" had the effect of building a reference domain with two itineraries, and "this itinerary" had focalized one of them. The second itinerary is therefore easily accessible. In the same way, replacing U4 with "I would like to see the direct itineraries": "the direct itineraries" is not interpreted in the set of all the possible itineraries, but in the reference domain defined by the previous utterance, "here are the possible itineraries". In this reference domain that only contains the journeys to Paris, the "direct" modifier allows the system to extract the set of direct journeys to Paris, which does match the referent desired by the user. Moreover, if we replace U4 by "I would like to see the itineraries to Marseille", the active reference domain, which contains the journeys to Paris and no journey to Marseille, leads to the identification of no referent, and thus a negative answer for the user, such as "there aren't any", or using the fact that the reference domain was built with regard to the Paris destination, "there aren't any, there are only journeys to Paris". This answer favors the common interpretation in the active reference domain, which is one of the operational

modes of this model. Another system could have erased the journeys to Paris and displayed those for Marseille, stating "here are the possible journeys" as in S2. But that solution consists of starting the query anew, and then ignoring the dialogue that has been carried out until then. It goes against natural dialogue. If the system chooses to consider the utterance U4 as a new query and thus build a new reference domain, then there is a referential rupture in the dialogue, and this rupture could warrant a request for confirmation, such as "for Marseille?", or a materialization of the rupture, such as "should I abandon Paris for Marseille?", uttered at the same time as the journeys for Paris disappear from the screen and those for Marseille appear instead.

6.1.2. *Visual scene analysis*

A first source to determine focus spaces or contextual subsets that match reference domains is visual perception. We saw in paragraph 2.1.1 that the HMD system knows the nature and spatial location of all the objects displayed on the visual scene so in our example, the train journeys which were highlighted graphically, and, in the case of a task such as the one carried out by SHRDLU, geometric shapes that create the microphysical world of the task. In this context, the user can focus on a subset, for example the shapes placed on the left or all the hollow shapes. This visual subset is determined using criteria such as those presented by the Gestalt theory: spatial proximity between objects, similarity, continuity, etc., see a ranking formalization in (Landragin, 2004) and (Landragin, 2006). It only becomes a reference domain from the moment when the user expresses a reference to a perceptive group (the group of shapes on the left) or a spatially isolated element or through its intrinsic properties such as size and color. This reference domain then allows the system to chart attentional focalization phenomena by allowing, for example, the interpretation of "the green block", not as the only object in the visual scene verifying the properties of being block shaped and green, but also the only object in the visual reference domain with these properties. In the case when the scene has another green block, which is not placed in the focus space, this mechanism allows the system to avoid a reaction such as "I do not understand which green block you are referring to", but instead it has the ability to model the attention and solve the references in a relevant manner.

An important phenomenon that can happen within this framework is that of visual salience, which allows the system to chart the user's attention focus on a specific object, when it is different from other visible objects due to specific properties: in the foreground, bigger, of a different color. In addition to the ability of automatically detecting perceptive groups, an HMD system relying on a visual scene displayed on screen thus benefits from automatically detecting visually salient objects, as we saw in paragraph 2.1.1. At the reference domain level, an object's salience does not contribute in building a new potential domain, but in focusing one of the elements of a reference domain built according to the perceptive groups. Thus, the exophoric pronoun, for example "it" without any possible linguistic antecedent, can be seen as referring to

the most salient object in the common reference domain. In this case, again, modeling multimodal reference domains allows for the system's in-depth understanding.

6.1.3. *Pointing gesture analysis*

In the case of a multimodal interaction with a touch screen, the user can make gestures, for example point or circle, to refer to the objects displayed. If the gesture trajectory perfectly matches the targeted objects, the HMD system can solve the reference without too much difficulty. If the trajectory is approximate, for example misses or involuntarily includes an object that is not part of the intentional reference, then the system is faced with cases of undecidedness. This is where the notion of reference domain can provide useful clarifications.

In general, processing a gesture can lead to the detection of an ambiguity on the intent behind the gesture: the same gesture with the same shape and same trajectory can come from various intentions. A hand movement tracked by a camera can, for example, match a paraverbal gesture that emphasizes a specific word but is not referring or can point at a specific object and thus carry a reference. The presence of a referring expression in the linguistic utterance as well as machine learning techniques applied to recognizing paraverbal gestures can obviously help solve this type of ambiguity. Once the system is sure that the gesture is a deictic gesture, the analysis of the gesture trajectory can in itself lead to detecting ambiguity. In the case of an interaction with a touch screen, one example is a gesture surrounding three objects but that also partially includes a fourth and ends very close to a fifth. Based on a structural analysis of the circling shape, i.e. a detection of the remarkable aspects of the trajectory such as points of inflection, crossings, constant curve areas or closing areas (Bellalem and Romary, 1996), of an analysis of the visual scene in terms of perceptive groups (paragraph 6.1.2), and potentially geometric index calculations such as covering level or relative distances, the system can then discard the fourth object, for example, because it is not part of the same perceptive group as the other objects in question, and since the gesture trajectory shows a slight avoidance movement when it covers the fourth object. However, it can decide to keep the fifth object as a potential candidate inasmuch as the fifth object is part of the same perceptive group as the three objects that are clearly circled. If the dialogue history has a reference domain that separates this object from the three which are circled, the decision made would have been the opposite. In any case, we now have two hypotheses: one about the three objects and one about the four objects. This is a pre-analysis of the gesture in visual context, and this pre-analysis will be confronted to the semantic analysis of the simultaneous oral utterance: thus, either a referring expression gives a number ("these three objects", "these four shapes", "this object, this one and this one") and the ambiguity has disappeared, or it has not. The ambiguity is then confirmed and the system must decide between choosing one of the alternatives or generating a clarification question for the user (Landragin, 2006).

Other ambiguities and analyses can be considered. Within the frame of an HMD relying on an HMI, any gesture is thus first considered as ambiguous, being either a conversational gesture meant for the dialogue system or a direct manipulation gesture meant for the HMI. At the analysis level, there are other approaches that consist, for example, of the gesture module not suggesting any hypotheses when faced with ambiguity (Martin *et al.*, 2006), which can lead the dialogue manager, if the linguistic module cannot find the referent on its own, to decide on a reaction without having any hypothesis (and thus to query the referent's identity). (Kopp *et al.*, 2008)'s approach, for gesture generation but the idea can be applied to automatic understanding, consists of implementing a gesture formulator that reasons based on a set of features with values translating localization, trajectory meaning, direction of each digit (when the hand configuration is tracked by a camera or a pointing glove), the palm orientation or even the hand's general shape. In general, the processes to be implemented depend on the modalities recorded in input: where the camera tracking requires many parameters to rebuild the meaning of an iconic gestures (or of a gesture in sign language), the user of a touch screen lowers the gesture interaction to be processed to simple trajectories, such as the classic interaction with a mouse.

6.1.4. *Reference resolution depending on determination*

The analysis of the visual context and that of gestures happens in parallel with linguistic analyses. As we saw in the previous chapter, they collaborate with each other to achieve a formal representation of the utterance's meaning that takes into account the prosodic, lexical, syntactic and semantic meanings of it. In "how long with this itinerary which seems shorter?", the prosody indicates, for example, a slight accentuation of the demonstrative referring expression "this itinerary". Syntax and semantics conclude with this prosody that "which seems shorter" is not a restrictive relative that could help identify the reference. The semantic lexicon allows the system to create a link between "itinerary" and the concept of travel described in the application's conceptual model, which turns out to be highly compatible with the fact that it lasts a certain amount of time, which is the aim of the utterance's main question. However, the referring expression "this itinerary" is not fully analyzed. There is notably no referent for it yet. To achieve this, the multimodal reference domain model determines an underspecified reference domain that translates the linguistic constraints carried by the referring expression. These constraints are first those of the words used, that is of the category and modifiers as filters to search for the referent among the available objects. In some cases, as in "color the pyramid red" or "delete this file", the verb's semantics and the sentence's semantics provide additional filters: the fact that it is not red for "the pyramid" and the fact that it can be deleted for "this file". Another constraint is that of the scope of determination. Depending on the demonstrative, the definite and undefinite's way of functioning, the referent research criteria are different. Thus, the demonstrative "this N" forces the focalization of the referent, either

by previously mentioning the same referent or simultaneously through a pointing gesture. As for the definite "the N", it operates by extracting the only N in the reference domain. As (Corblin, 1995, p. 51) states, "the N" consists always of opposing a previously mentioned N to other entities to predicate something about it. It has to oppose the element that is an N in the reference domain to elements that are not N. Finally, the undefinite operates by selecting a random element from a set. These three cases are far from covering all the referring expressions allowed by language (plurals have their own mechanisms as do personal pronouns and proper nouns), but they illustrate the three main mechanisms involved in reference resolution: using focalization, extraction, selection.

At this point, the HMD system thus has an underspecified reference domain with linguistic constraints on the referring expression, and this underspecified domain will try and work with the reference domains provided by the visual context, by analyzing the gesture if there is one, and by the dialogue history. One of the roles it has is to save the successive reference domains, so as to chart the phenomena or contextual broadening, contextual narrowing, anaphora, as well as alterity, with expression of "other" such as "the other trains" or "the other pyramid". Depending on the task, the matching can be tested in a specific order, by favoring, for example, visual perception over the dialogue history and stopping as soon as a complete result is achieved, or it can consist of clarifying all the possibilities, so that the dialogue manager can decide which alternative to choose if there is any ambiguity. With our example U2 involving the referring expression "this itinerary" and a pointing gesture toward one of the itineraries displayed on the screen, we are in the simplest configuration possible: the underspecified reference domain imposes an existing focalization in a domain that covers the different itineraries to get to Paris, the gesture provides a hypothesis on the itinerary pointed at, and the matching leads to the consideration that the focalization is on this hypothesis, and thus to a reference resolution. In other more complex cases, we might need to determine the type of access to the referents with a fine analysis of the access type and determiner type combinations (Landragin, 2006). In any case, a formalism like feature structure allows the system to implement such a model, and the matching is carried out by a unification operation. The challenge for HMD is thus mostly to determine all the types of reference, to write a module in charge of deducing from the linguistic forms the formalized constraints in the reference domains, constraints which will direct the feature structure unification.

Moreover, the utterances that contain more than one reference can cause problems linked to multimodal fusion. In an example, such as "is this itinerary longer than these and these?", three referring expressions can be the focus of a pointing gesture, or even several pointing gestures for "these". If the system receives five pointing gestures, an in-depth analysis of the temporal synchronization and matching possibilities between gestures and expressions is necessary to determine which gestures are linked to which expressions. The only constraint due to the natural use of language and gesture is that the successive order of the gestures follows the successive order of the expressions.

In the extreme cases observed for tasks leading to a number of references (Landragin, 2004, p. 45), the combination can become so complex that heuristics are required. These phenomena lead us particularly to distinguish between various levels of multimodal fusion. Where many approaches focalized on signals carry out a matching of gestures and expressions only based on the temporal synchronization setting, i.e. by running a physical multimodal fusion, other approaches such as those of reference domains highlight another level of multimodal fusion: the semantic level (Martin *et al.*, 2006; López-Cózar Delgado and Araki, 2005). Chapter 7 will present a third, pragmatic, level related to dialogue acts.

Reference can also be applied to entities other than concrete objects such as pyramids and itineraries. In the classic example "put that there" (Bolt, 1980), there is a first multimodal reference that does concern a concrete object, linguistically referred to as "that" (i.e. one of the vaguest referring expressions there is in English) but also concerns a location, with "there". This second reference is necessarily accompanied by a gesture and is thus a case of multimodal reference. The resolution of this reference creates other issues than those we have seen until now. The nature of the referent is indicated by "there", but the exact determination of the referent depends on several parameters: nature of the action, here a positioning; nature of the object to be placed, especially its size (putting a nail in a particular place is not remotely similar to putting a carpet in a particular place, according to the example given by L. Romary in the context of an interior decoration task); and nature of the objects already present in the indicated location (see following section for the setting linked to the action). Moreover, the reference can be applied to abstract objects, which can be concepts known to the task, for example "delay" or "cost" as in "delays are unacceptable" or "what is the cost of this journey?", or states, actions or processes, as in "being late is unacceptable" or "how much does this journey cost?". In general, any full word can have a reference. In any case, the reference resolution process can take inspiration from the detailed process for concrete objects, or the one we will now see for actions.

6.2. Action reference resolution

The example "put that there", beyond the fact that it has two multimodal references, refers to an action carried by the verb in the imperative mood. Depending on the task and the possibilities it presents, a link between the word "put" and one of the actions that can be carried out by the application is not necessarily easy or self-evident. For this is the center of the issue: given an utterance, what function of the application is it referring to? Using the example of speech-based drawing software, "put that there" can trigger an action of object movement, an action which is more linked to the verb "move" than the verb "put", the latter also being able to trigger the creation of an object ("put a block there"). Solving the reference thus involves the semantics of the verb used, its valency (paragraph 6.2.1), as well as the objects in question, and, in general, the task in progress (paragraph 6.2.2).

6.2.1. *Action reference and verbal semantics*

The task model groups the list of actions that the application can run. Solving a reference to an action thus consists of bringing it back to one of the elements on this list. For such application elements, (Duermael, 1994) uses the term of operator, and defines it as an action model, made up of preconditions, postconditions and a body. The body matches a function of the application. The indispensable preconditions enable the system to check the applicability of this function by checking, for example, that the objects in question are indeed compatible with the function. The postconditions are used to simulate the function's running, just before carrying it out: the goal is to simulate the effects with transient representations of the objects and the knowledge, so as to see what the objects and knowledge undergo and what the result is. This anticipation allows the system, however, to detect problems that are difficult to predict, for example collateral ones, and important, such as the deletion of an object. If the system believes it is relevant, it can then warn the user of the consequence and ask for confirmation. The anticipation also allows it to implement a dynamic management of the action, and implement a cancellation function, which is not necessarily easy when deleting and modifying important objects.

An application function requires parameters. It is often applied to objects according to very specific properties. An object movement function thus requires knowing the object in question and the considered location, as well as the preconditions on the fact that the object can be moved, and that it will fit at its destination. To run, the function needs these two parameters. In the simplest case, the user's utterance has a verb whose semantics are clearly linked to that of the intentional operator and whose valency matches the number of parameter required. This is the case for "move that there": since the two parameters are not of the same nature, the action's reference resolution happens very simply. In other cases, for example with "I am moving that there" or in "move that", the reference resolution requires the system either to ignore a parameter, since it is not in fact a parameter but only a neutral expression with regard to the task to be carried out, or to detect the absence of a parameter, which leads the system to query it. This is obviously the most common case in train ticket reservation dialogues (and also why thematic roles are identified during the semantic analysis), which the U1 example highlights: "I would like to go to Paris" does not mention the departure station or the departure time, which are necessary parameters. Faced with this utterance, the system can consider that the user is starting a query which has to be completed by the dialogue together with the system. The system can then *plan* the precision queries and start with the departure station, as we saw in paragraph 3.2.1. It can also try to deduce the missing parameters by referring to the dialogue history (if a departure station was mentioned at any time, it is a relevant candidate), to the situational context (the departure station matches the location of the terminal used for the dialogue) or to common sense (someone trying to buy a train ticket might want to leave immediately, and this is a choice that the system can suggest before asking the next question).

As always when it comes to language, ambiguities can occur and complicate the reference resolution. In a system such as SHRDLU, this is the case when we want to put one object on another or slot one in the other: an utterance referring to an action requiring two objects as parameters can lead to two interpretations, the correct interpretation and the interpretation in which the objects are switched around. To decide, an analysis of the prepositions used and the thematic roles is essential. More complicated is the utterance such as "reserve a train ticket for Paris immediately": does it mean as soon as possible (on the next train) or does the "immediately" complement only refer to the reservation order? A sentence can have optional complements in the verbal valency meaning, as well as intermediate elements, which cannot be predicted with the verb used but only by looking at its hypernyms, unnecessary elements, which cannot be predicted with the verb since they match adjunct adverbials or extra-peripheral elements such as logical or discursive modifiers, e.g. "you know". One of the HMD's tasks is to use these linguistic elements to better manage the running conditions and parameters of the application functions. Beyond the linguistic aspects, it also happens that the action reference resolution involves purely applicative aspects. If we consider, for example, that the task is implementing an object deletion function, which only works with one parameter, thus a unique object, then the utterances such as "delete these objects" lead either to an error message ("I need a single object. Which one shall I delete?") or to the implementation of a chain of executions of the deletion function. The last solution is not necessarily relevant, for example if the deletion leads to consequences for the other objects.

To solve such examples, reference resolution can involve a temporal model. The goal is to take into account the temporal constraints involved in the running of a function, which allow the system to very accurately model chain actions, as well as the interactions between carrying out the actions and the parallel evolution of the world of objects. With the train ticket reservation example, such a temporal model is all the more necessary if there are several terminals allowing several users to reserve tickets for the same trains (Kolski, 2010). The point of a temporal model can also be found in a better use of the verbs' linguistic characteristics: semantic classes, aspect classes (inchoative or non-inchoative depending on the hypothetical beginning of the action, terminative or non-terminative depending on its end), past participle roles or even preposition roles. On certain points, the HMD system still has a long way to go.

6.2.2. *Analyzing the utterance "put that there"*

The example "put that there", which sparked the beginning of multimodal dialogue (Bolt, 1980), can be broken down in more detail when the following figures are taken into account, all within the framework of speech-based drawing software use:

– The reference "that" refers to a static object, which is part of a graphic palette and cannot be moved. In this case, "put" refers to an action of generating an identical copy and not a movement.

– The reference “that” refers to an object that is not stored in its correct place (i.e. where all the other copies of the same class of objects are), or which is not in the right configuration or direction. In that case, “put that there” might be more than movement: it can also be a rotation or a sorting according to the settings of the objects already sorted.

– The reference “that” depends on the result of the reference resolution “there”: the location referred to by “there” might, for example, be a place to sort objects of a certain category, and the accompanying gesture “that” is potentially ambiguous between several objects of different categories.

– The reference “there” depends on the result of the reference resolution “that”: it is the difference between “put carpet there” when pointing at an area of the room, and “put a nail there” with the same gesture.

– The reference “there” depends on specific knowledge, for example when “that” refers to an electric plug: the gesture accompanying “there”, even if it is precisely carried out, cannot point at the exact location of the plug, since it has to answer to height standards, or distance from another plug standards, which take priority when placing it and lead to reinterpreting the gesture as an approximate pointing gesture.

– Both references can be generated at the same time as a single gesture, which might describe a (movement) trajectory or could be solely interpreted with the two extremities (disappearance and reappearance). The ambiguity could be important, for example when the movement action leaves a visible track on the screen, a track that itself becomes a part of the drawing in progress. In the movement hypothesis, additional ambiguity can happen if the task implements two types of movement: one with no effect on the objects found on the path it describes, and the other leading to pushing any obstacle away.

These examples show how important action modeling and natural language underspecification are. To solve them, it is useful to implement a multistage reference resolution process that contains:

– visual content analysis (analysis of the perceptive groups and the differences between “that” and the objects already “there”);

– analysis of the gesture trajectories (presence of an avoidance phase, for example);

– linguistic analyses (verbal semantics, thematic roles and temporal aspects);

– confrontation of the three analyses thus carried out to solve, in a parallel manner, the object references and action references (multimodal fusion that takes the constraints of each modality into account);

– confrontation of the pragmatic analyses, which will be the focus of Chapter 7.

6.3. Anaphora and coreference processing

We mentioned one of the roles of dialogue history, which is to remember the mentioned objects as well as the expressions used to point these objects out, as the dialogue progresses. Thus, it is possible to solve the anaphora and identify the coreferences. Solving anaphora is a process that is much studied in linguistics and NLP (Mitkov, 2002), which consists of creating a link between an anaphoric expression and its antecedent. Thus, in "take a cube and put *it* in the box", "it" is an anaphoric expression, i.e. it cannot be interpreted in the immediate visual context but requires the system to explore the linguistic co-text so as to find a referent that has already been mentioned and is taken up again. Searching for the antecedent leads to identifying "a block" and building an anaphoric relation between "it" and "a block".

In an HMD system, this process requires various stages. First, we need to identify the genuinely anaphoric expressions and distinguish them from those that are references like the ones we have seen so far in this chapter. To this end, the linguistic form is essential, third person pronouns clearly favor an anaphoric interpretation, whereas "the block" and "the green one" can refer to something either directly or anaphorically: "take a red block and a green block, put the red one in a box and the green one on top of it". The impossibility to solve the direct reference is also a hint: if various referents are possible, it might be an anaphora. A second stage consists of looking at the gender, number and potentially category if the anaphoric expression has a head noun, so as to draw up a list of potential antecedents. When various antecedents have been identified, a choice then has to be made. The criteria on which this choice relies are proximity, for example in terms of the number of words between the antecedent and the anaphora, and salience of the referent matching the antecedent or grammatical functions: if the antecedent function is the same as that of the anaphora, there is a syntactic parallel and it is an argument in favor of that antecedent rather than another antecedent. In NLP as well as in HMD, this process can be implemented through statistics, or even machine learning, so as to weigh the importance of each resolution setting versus the tests on the corpus. In HMD, the antecedent can belong to a previous utterance, and the speaker's identity is not a limit: the user can anaphorically take up a reference made by the system, and vice versa.

Until now, our anaphora examples are also coreferences, i.e. the antecedent and the anaphoric expression refer to the same referent: once the relation between them is identified, the allocation of a referent to an anaphoric expression consists of taking up the referent already allocated to the antecedent. Yet, the anaphora is particular since it can be associative, i.e. use a conceptual link between two different referents. Thus, "give me a ticket for Paris. *The cost* must be less than twenty euros" or "draw a triangle. Color *one side* red" involve two referents linked between themselves every time, the reference to the second understood through the use of the first, by an associative anaphora relation. The anaphora is not linked to coreference in that case: the

two referents are not the same, and each of them requires its own reference resolution process.

As for the reference, the anaphora and coreference can refer either to concrete objects or to abstract objects, and especially events. In "the reservation failed, I did not get a ticket", there is a link between the reservation's progress and receiving a ticket. In "I have reserved a single ticket for Paris with a specific time and seat. I will start again with a single ticket for Lyon", the verbal phrase "to start again" can only be understood with the antecedent that clarifies the reservation. Finally, an example of event coreference within the frame of our favorite task is the following: "I am reserving a ticket for Paris. I would like a single". In all these examples, the HMD system is confronted with two sentences or two propositions that both describe an event, and that are linked to one another. The link depends on the nature of the events and their representation in a conceptual model. Thus, getting a ticket can be considered as the last constitutive step of a reservation. The challenges for an HMD system are multiple in this case: it first has to determine the anaphoric or coreferring link, then to link the semantic contents of both sentences using this link and then to infer a semantic content that could cover both sentences, or if that is impossible, to clarify the type of discourse relation that exists between them. These are the key aspects to understanding, which allows the system to approach a dialogue's coherence with efficiency, but involves many pieces of knowledge that are hard to implement. In the case of the first example, the second sentence "I did not get a ticket" explains the observation made in the first sentence. To identify this discourse relation, the system has to understand not only the link between both events but also that the user is expressing his problem with a reasoned description. The system can thus refute the link by answering "the reservation succeeded but the ticket was sent to you yesterday and will only arrive tomorrow". In the case of the second example, the semantics of the verbal phrase "to start again" allows the system to start a new reservation by using all the settings of the old reservation except that the destination is clarified. In the case of the third example, the two events are simply one and the same (but this has to be understood first), which allows the system to define a query with all the settings, those mentioned in the first sentence and those mentioned in the second sentence.

As we can see, reference is indeed a pragmatic question that goes beyond the simple identification of a referent: with the notion of reference domain, with the use of dialogue history and the links that are drawn between the different modalities and with the notions of coreference and coherence, it appears to be a complex mechanism that contributes to the dialogue's coherence.

6.4. Conclusion

The links between a linguistic utterance and the world of the task to be carried out are made through references: on the one hand, references to objects that are accessible

and can be manipulated in this world, which can be done because of the well-chosen words and expression, or because of the combination of a pointing gesture with an appropriate linguistic expression (in the case of multimodal systems), and on the other hand, references to various actions that can be carried out in the world of the task. This chapter outlines the reference resolution processes for references to objects and actions, and shows how various utterances can be linked to one another by anaphora and coreference relations that a system has to identify to be able to manage the dialogue's coherence.

Chapter 7

Dialogue acts recognition

The utterances “reserve a ticket for Paris”, “how long is this journey?” and “I am unable to communicate with you” are not only different from a semantic content point of view, but also by the speech act they carry out: the first is an order (“telling to”), expressed with the imperative mood, and requiring the system to carry the order out; the second is a query (“asking”) expressed in the interrogative form and requiring an answer from the system; and the third is an assertion (“saying that”) expressed as a declaration and requiring the system to take what is said into account and come to a conclusion, whatever it is. The nature of these speech acts and their identification mechanisms, classified here according to the relevance theory point of view (Sperber and Wilson, 1995), are an aspect of pragmatics (called third-level), see paragraph 1.2.2, which is essential when managing dialogue: it is by understanding what speech act the user is carrying out that the HMD system can determine its own reaction. At least, and we will see this in this chapter and the following which is linked to it, this is one of the parameters which allows the system to decide how to progress with the dialogue.

Just like for the action model described in the previous chapter, achieving a speech act involves preconditions and postconditions, and its identification requires a certain number of parameters. In automatic understanding, the process responsible for the identification of a speech act requires input arguments which can be preprocessed, and provides results at the output. At input, we need here the semantic representation obtained in Chapter 5, with the prosodic indications, especially those concerning the intonation outline of the utterance, and with the reference resolution results obtained in Chapter 6. We also need the dialogue history with the semantic and pragmatic representations calculated for the previous utterances, including the identification of speech acts which were carried out. Finally, in the case of multimodal dialogue, we

also need a representation of the gestures carried out, and in general a representation of the content carried by the modalities which are processed to potentially allocate them a speech act which, due to the nature of these modalities, is called a *dialogue act* rather than a *speech act*. We will see that a gesture can indeed express an order, a query or an assertion.

The result of the recognition of dialogue acts is the affectation of one or more labels to the semantic content, these labels described the dialogue acts which were carried out. As always, several alternative hypotheses can be generated in the case of ambiguity and an underspecified representation in the case of an impossibility to recognize a specific act. The pragmatic representations thus obtained are the main parameters for the dialogue management and the determination of the system's reaction, and they also update the dialogue history.

This chapter is organized as follows: first, section 7.1 aims to describe the nature of the dialogue acts; second, section 7.2 presents a few methods used in HMD for their automatic identification, especially the processes implemented in the multimodal dialogue, with the multimodal fusion process at the level of dialogue acts (section 7.3).

7.1. Nature of dialogue acts

7.1.1. *Definitions and phenomena*

(Austin, 1962) gives each utterance a locutionary act, which corresponds to the generation of the utterance; an illocutionary act, that of querying, order, etc.; and a perlocutionary act, which is the often the intentional generation of certain effects on the beliefs and behaviors of the hearer. The term speech act is used to describe the illocutionary acts. (Searle, 1969) extends J. Austin's categories and characterizes five types of main act based on the criteria such as the sincerity condition or the direction of fit of the act, i.e., does it act on the world or is it the opposite: assertives; directives, whose goal is to make the hearer do something and in which we find queries and orders; commissives, which commit the speaker to a future action; expressives and declarations. This theory is the focus of many variations and adaptations, for example that by (Clark, 1996) who considers the following types of act: assertion, order, closed question, promise, offer, thanks, compliment, greeting and farewell. In general, the different approaches and criteria that lead to the determination of one list of acts rather than another are presented in (Traum, 2000).

(Sperber and Wilson, 1995), within the context of their cognitive and pragmatic approach to dialogue, suggest to abstract the categories into "telling to" (which we will call order), "asking" (query) and "saying that" (assertion), which focus on what the system has to identify to interpret the utterance. These three types of speech act are not based on the utterance's syntax, as the beginning of the chapter might have

led us to believe, do not involve conditions as J. Searle's do, and, at least for humans, can be identified thanks to simple linguistic tests: adding "please" allows us to test the order, adding "tell me" at the beginning of the utterance allows us to test the query and adding "after all" allows us to test the assertion. The syntax provides hints without any syntactic structure being linked to a type of act: "may I have a ticket for Paris?" as a query, mostly satisfies the test of the "please" characterizing an order. Prosody also provides hints, with, for example, a rising (or rather a constant, but not a sinking) intonation outline allowing us to interpret "you have tickets for Paris" as a query rather than an assertion. As for the last example, the act of querying, especially if the prosody is not very strong, can, however, not be visible, at least much less than it is in "do you have tickets for Paris?". Because of this remark (Kerbrat-Orecchioni, 2012), we could find a distinction between the illocutionary value and the illocutionary force: in both cases, the value is a query, whereas the force is rather weak in the case of the assertive form and strong in the case of the querying form. This allows the system to better characterize the utterance's speech act, and thus react in a relevant manner.

Beyond speech acts, the dialogue implements acts linked to the progress and modalities of communication. We have already seen that the possibility for gesture acts has led us to talk of *dialogue acts*. This term is also used to refer to acts which can be understood in a dialogue context, that is, taking the previous utterances into account. Thus, due to its limited semantic content, an utterance such as "yes" is allocated as an assertion speech act, but is more precisely modeled by a dialogue act of an acknowledgment type or answer to a query type when we consider that the previous utterance is either "I want a first class ticket" or "are there any seats left in the next train for Paris?". Thus integrating dialogue aspects into the notion of an act can be problematic, and some authors refuse this vision, considering that links between utterances are carried out at a different analysis level, in dialogue management (see Chapter 8). Nonetheless, it remains that a speech act is understood in a dialogue, which (Grisvard, 2000, p. 102) shows with the example "you erase the sequence", which, preceded by "what happens if I press OK?" can be interpreted as an assertion, whereas if it is preceded by "OK, so what do I do now?", can be interpreted as an order. This type of example shows why it is indispensable to refer to the dialogue history. Finally, dialogue acts can be considered as constituting a category of *conversational acts* (Jurafsky and Martin, 2009), with the category of speech turn acts, grounding acts (acknowledgment, acknowledgment request, repair, repair request, continuation, cancellation, etc.) and argumentation acts (elaboration, clarification, opposition, etc.). This more global point of view may become broader by integrating the possibility of collaborative acts, that is, carried out cooperatively by the speaker and the hearer, such as when an utterance by the first completes the utterance by the second.

In the end, an utterance such as "how long with this itinerary which seems shorter?" can carry several conversational acts: not only explicit acts such as the query on the journey time and the comment on the fact that it seems to be shorter (paragraph 7.1.2), or simply the fact of speaking following the dialogue system's act, but also tacit acts

such as the one corresponding to a grounding of the utterance "here are the possible itineraries": it is because the user understood this utterance well that he can allocate the status of alternatives answering his initial query to the graphical elements appearing on the screen. Moreover, the act of acknowledging the previous utterance is also a tacit act involved here. We then reach the notion of *multifunctional act* (Bunt, 2011), which we will refer to later as a composite act (paragraph 7.1.3).

7.1.2. *The issue with indirect acts*

In the example of the introduction mentioned above, "which seems shorter" is a relative clause which can be interpreted as the proposition "this itinerary seems shorter". With its assertive form, the comment made by the user can appear to have little importance, or at least not to concern the system but be more of the nature of a passing thought. The only linguistic test that works with this proposition is that of the assertion: "after all, this itinerary seems shorter". However, with this comment, the user might be trying to get the system to react as if it were a query: "this itinerary seems shorter, right?". The phenomenon described here is that of an *indirect* speech act. It has been greatly studied from a theoretical point of view (Searle, 1969; Chapter 5 of Levinson, 1983; Moeschler, 1985) and also from a more formal point of view (Searle and Vanderveken, 1985; Asher and Lascarides, 2001). Some authors consider that the speech act carried out by the utterance is an assertion, and our way of interpreting it turns it into a question. Others consider that one form can take the place of another by convention. This is notably the case for an order, which, when given in the imperative, can appear brutal, to the point that the query is prefered (and becomes a conventional indirect act). Others, and this is especially the case of (Asher and Lascarides, 2001), put the current utterance into perspective with regard to the previous or the following, then clarify the discourse relation that exists between them and help the analysis fall down on the side of the indirect act. More specifically, the discourse relation itself becomes an act, and the query which looks like an assertion becomes a complex act, which does not cause the HMD system any problem because it is already used to manage all kinds of ambiguities. Whatever path is chosen for the interpretation, the important part for the HMD progress is that the system must understand that it can react by answering the comment, that is by confirming or denying that it is the shortest itinerary. If the system does not detect this *indirect* interpretation possibility, then it can only answer the question asked by the main proposition "how long with this itinerary?". This might not be an issue, but it does lower its understanding abilities and cooperation abilities a great deal.

Another typical example of the indirect act phenomena is the query hiding an order, such as "can you listen to me?" in our train information task. With the linguistic tests seen at the beginning of paragraph 7.1.1, we can see that the order and the query both work well with this example: "can you listen to me please?" and "tell me, can you listen to me?". The hypothesis of the simple question does not hold water very

long: the system is absolutely able to listen to the user, so the answer "yes I can" does not bring anything to the dialogue. Unless it is a specific case, as if the user has just approached the system and not yet noticed its understanding abilities, this is not what is at stake here. The question "can you listen to me?" could also refer to the communication situation conditions: in a noisy environment, the user may believe that the system cannot hear him ("can you hear me?" would in this case be more relevant). Realistically, it is an injunction that the user is carrying out, or even an overtone such as "pay attention to what I am saying, will you" or "I am telling you I want to go to Paris, not that I am looking for a promotional sale for my next vacation". If the overtone is hard to identify, the order intention is easier to infer, and we can then consider this utterance as an example of an indirect act which can lead the system to react by saying "I am listening" or "please restate your query".

Compared to the example "this itinerary seems shorter", it is, however, possible in this case to answer the question at the same time with "yes, I am listening" or "yes, please restate your query". In other words, we can consider that the question and the order happen simultaneously, which is a *composite* act.

7.1.3. *The issue with composite acts*

Beyond the case in which an utterance both acknowledges the previous utterance and the speech act carried by the linguistic form, the case of composite acts is interesting as soon as a speech turn has various speech acts. In some manner, this is the case for "hello, I would like to go to Paris" and "OK, I will reserve a ticket", because both utterances actually have two discourse segments, and a speech act is allocated to each segment. Determining speech acts thus comprises a segmentation of utterances as a preliminary. This is especially the case for "how long with this itinerary which seems shorter?", which is interesting and falls under the composite act classification, with a "asking" component, which is direct and concerns the journey time, and another "asking" component, which is indirect and takes the shape of the assertion "this itinerary seems shorter". According to some authors, the first component, that is the main component from the point of view of sentence construction, is called the *main act* or *act of first intent*, while the second component is called the *subordinate act* or *act of second intent*. As for the indirect act, the assertion component, that is which matches the linguistic form, is called the *surface act* or *secondary act*, while the querying component, that is which is closed to the communication intention or at least the act to which the system should react, is called the *profound act* or *primary act*. Another example of the plethora of terms, (Kerbrat-Orecchioni, 2012) also distinguishes between the patent (literal, explicit and primitive) and the latent (indirect, implicit and derived) values with various cases in which the real intention is the latent value.

Here we have analyzed "how long with this itinerary which seems shorter?" as a single discursive segment, which is thus affected with a composite act with two aspects. Another possibility would have been to consider two segments, along criteria which have less to do with syntax (there is only one sentence) and more to do with function or communication, and we invite the reader to refer himself to functional segments in (Bunt, 2011). The result is the same with both approaches: the system is confronted with two acts, and it can react to one or the other, or even to both. The advantage of the solution which considers the utterance as a single segment is that it emphasizes one act over the other, which is harder to do if it considers two autonomous segments. The direct act matching the question in the main clause is emphasized compared to that of the comment, due to the fact that it is linked to a main clause and not a subordinate or relative clause. Thus it is mostly syntactic criteria that take precedence here. There are, however, situations in which a single segment clearly leads to the identification of a composite act. This is the case for the utterance "what about eight o'clock?" as an answer to the query "do you have a train for Paris tomorrow morning?". It is a query from the system which relates to a time, in reaction to the user's query. Yet the fact that a time is mentioned proves that the system has understood the query: not only does it answer in the affirmative (if the answer had been negative, this utterance would not have been possible), but it also requests a precision to validate the reservation. We thus have a composite act or, to take up the point of view in (Asher and Lascarides, 2001), a complex act including an elaboration relation between the two utterances.

Without questioning the classification into three main speech acts which relevance theory establishes, this type of example shows that natural language does not fall into totally compartmentalized categories. The interpretation suggested here for the introduction example is more limiting. To do the job properly, we would have to integrate the aspects of the speech turns, of the grounding which would take into account the previous utterances and, in general, of all the communication functions which occur in a dialogue, such as the management of time and the task in progress.

7.2. Identification and processing of dialogue acts

7.2.1. *Act identification and classification*

In HMD, determining the nature and typology of speech acts as well as dialogue acts must be able to cope with two potentially contradictory goals. On the one hand, it has to determine a precise classification of the types of act so that it becomes the system's main reasoning criteria: as the number of the types of acts increases, the automatic identification can become problematic, but the interpretation also becomes all the more detailed and the system can react with all the more relevance. In general, a classification with a reasonable amount of types is more practical for the system to manage. It has to consider a corpus annotated in terms of dialogue acts to use these

corpora to improve the system identification performances, for example, thanks to a machine learning phase. Yet building a reference corpus on these aspects is still done by hand for now, and it is difficult to ask annotators to choose, for each utterance, one type of act among a hundred. The annotation task becomes too onerous and could lead to a number of errors. Moreover, it is far from easy for a human to conceptualize a hundred types of act, and it does not help the system designer.

Various technical act classification suggestions have been made, from Foundation for Intelligent Physical Agents (FIPA) to Dialogue Act Markup in Several Layers (DAMSL), from an XML-based language such as KQML to the recent ISO 24617-2 standard specified by a work group bringing together the main international HMD research specialists. The point, especially for the latter, is to suggest a hierarchical classification that allows the system to approach the types of act at different levels of granularity; see also the summary table in (Harris, 2004, p. 104). This might be the solution to the previous paragraph's dilemma: in the case where an HMD system can exploit all the levels of granularity, a corpus annotator will make do, at first, with the first level.

The identification of the speech act or dialogue act is a process requiring the following parameters:

– the words of the utterance themselves and their semantic properties, especially for the verb (Rosset *et al.*, 2007);

– the syntactic analysis of the sentence, especially the mood;

– the prosodic analysis of the utterance, especially with the intonation outline (Wright-Hastie *et al.*, 2002);

– the type of the previous act in the dialogue, and in general any information arising from the dialogue history that can help link the current utterance to the previous utterances;

– a chain of the previously expressed acts, like in a language model, and to use methods such as CRF and HMM (see paragraph 2.1.3).

(Jurafsky and Martin, 2009, p. 880) emphasize that the module devoted to act recognition can be divided in two parts: in charge of general acts and in charge of recognizing specific acts such as corrective acts on the user's part following a system's error. Both parts operate in the same way, that is in the current systems they operate according to a labelling task after a machine learning phase, but with different models. The corrective utterances are indeed harder to recognize than the regular utterances and require specific indications such as the presence of words like "no" or "I do not have", the presence of a repetition, potentially with an exaggerated articulation, a paraphrase, an addition or omission of content.

7.2.2. *Indirect and composite acts*

The identification process of indirect acts takes up the same parameters as those presented in the previous section, but it also emphasizes four additional aspects which are particularly important:

– An index of dialogue conventions, which includes a few typical examples of indirect acts: if the situation being processed matches one of the examples, then the system can rely on the suggested solution.

– The speaker's preferences, that is the user model, if it has been updated as the dialogue progresses, especially in cases of indirect act detection ("when he says this it is to do that").

– The system's task model, and especially the list of its abilities: this is indeed a means to identify indirect acts such as "can you hear me?".

– The hypotheses on the user's mental states, following, for example, a BDI model (see paragraph 2.1.2), so that the system can detect when the user already knows the answer to the query it is making, to interpret it as an indirect act.

In the same way, composite act identification requires a similar list of parameters with a few changes:

– A set of words and linguistic constructions which are often used to express an act of second intent: epithet adjective, evaluating adverbs, appositions, relative clauses, etc.

– An index of dialogue conventions, which includes the examples of composite acts with a set of possible reactions for each act: for example, we react to an act that has a question and a comment by answering the question, and maybe by confirming or denying the comment (especially if it is wrong).

– The speaker's preferences, which can be paraphrased as "when he says this it is to do this and that".

– The task model so as to determine the order of importance of the different acts present.

– Hypotheses on the user's mental states, so as to favor one act whose satisfaction will have the greater incidence on these mental states. We here clearly fall back into the relevance theory with the notion of contextual effect (Sperber and Wilson, 1995).

In both cases, the process implemented in the current systems once again relies on machine learning. For these indirect and composite acts which are both explicit and implicit, there is an adapted technique, which is supervised classification, with act labels as hidden classes to be detected (Jurafsky and Martin, 2009).

7.3. Multimodal dialogue act processing

We have seen in paragraph 5.1.4 that certain gestures can carry a dialogue act. In the case of an HMD system with camera tracking, wide-opened eyes can be equivalent to a question such as "what did you say?". If they accompany a pointing gesture, the communication intent might be something like "what is it?". A pointing gesture can also match an order, like an X-shaped gesture on an object can mean the order to delete this object in the case of a touch screen system. Finally, and this is the case of most of the expressive gestures, a gesture can carry out an assertion whose content is added to the simultaneous linguistic utterance's content according to the multimodal fusion process at a semantic level (see paragraph 6.1.4). Obviously, if there is no linguistic utterance accompanying the gesture, its interpretation finishes with the determination of its single dialogue act. However, in multimodal dialogue, we are faced with examples in which linguistic utterance is given a speech act, as is the gesture, whether it is a "telling to", an "asking" or a "saying that" (we will remain once again here within the framework of relevance theory) and in which the automatic understanding goes through the processing of these multimodal dialogue acts.

The process to carry this out is that of a multimodal fusion at a pragmatic level: both acts, that of the gesture and that of speech, are confronted and unified to obtain a single act characterizing the complete multimodal utterance. First, when the two acts present are of the same type, for example two assertions or two questions, multimodal fusion essentially consists of checking the compatibility of the semantic contents. In the example given previously, wide-opened eyes do not carry a specific semantic content when this gesture happens at the same time as an oral utterance. The fusion is therefore immediate when the oral utterance is of the "asking" type. It is the same thing for a sudden gesture which illustrates, in an injunctive way, the order also transmitted through a simultaneous oral utterance. Fusion can be less immediate with two assertions: in that case, the gesture carries semantic content, for example that of a specific quasi-linguistic aspect. Either this content is compatible with that of the assertion stated through speech and the multimodal fusion obtains a single assertion act with a unified semantic content, or the two semantic contents are not compatible and the system is faced with two different acts, also known as a *multimodal composite act*. Finally, when the two acts present are of different types, for example a querying gesture with an oral utterance of the "saying that" type, there is more than one possible case. Either the semantic contents can fuse, and the HMD system can then emit the hypothesis of a *multimodal indirect act*: the linguistic utterance looks like an assertion, but taking the gesture into account questions this interpretation and suggests that of the profound "asking" act. If the semantic content lends itself to this the hypothesis is kept and the gesture then has the same exact role as a question intonation outline. Or the semantic contents do not fuse and in that case we are faced either with two distinct acts or with a composite act which comprises a query with its semantic content and an assertion with its own semantic content. We can then consider a ranking operating between the two: the linguistic act takes precedence over the gesture act if

only because a dialogue is first and foremost linguistic. As in the case of the example "how long with this itinerary which seems shorter?", the system will have to decide on its reaction while taking into account the three possibilities that it is presented with: react to the first act (in this case the linguistic assertion), react to the second act (in this case the querying gesture) or react to both. This is part of the dialogue strategy and is the focus of the next chapter.

7.4. Conclusion

When a human-machine dialogue system is used, it is often less with the aim to discuss on an equal footing than to get the machine to carry out a task. An utterance thus carries a query, an order or a question. These dialogue acts, which can be indirect or not explicitly indicated by the utterance, must be correctly identified by the system. It can then determine what type of reaction to adopt and approach the dialogue as a structured succession of significant acts. This chapter shows how the dialogue acts are processed and gives examples of complex acts, on the one hand at the level of language, which is naturally complex, and on the other hand at the level of interactions between language and communication gestures.

THIRD PART

System Behavior and Evaluation

Chapter 8

A few dialogue strategies

Dialogue management is often presented as the heart of the HMD system. This is where all the machine understanding process results arrive, where all the different information is compared, including that stored as the dialogue progressed, where decisions are taken, which might involve a problem resolution phase or a database information research phase, and where the messages for the user are generated as an abstract form which is then materialized by the modules in charge of automatic generation and synthesis.

As with each chapter of Part 2 of this book, there is a process that requires input arguments and returns output results. The first set of arguments covers the prosodic, lexical, syntactic, semantic and pragmatic analysis results, in the form of a representation of the meaning of the utterance, with all the references being solved, and accompanied by a label pointing to the dialogue act carried out. A second set of arguments covers the following resources which are independently managed or linked (or even at an internal level) with the dialogue manager: model of the task, objects and world in which the task is carried out, dialogue history and user model. The visible result of dialogue management is the system's choice of reaction. This reaction can be an action in the world of application objects, an action which is thus displayed on screen. It can be the visual presentation of information. It can also take the shape of a linguistic utterance. In that case, the dialogue manager returns a message in a form that indicates *what to say*, without going into the details of *how to say it*, a process which the automatic generator takes charge of with the module dedicated to ECA, if the system is visually represented by an avatar, and by the module dedicated to multimodal content presentation (see Chapter 9). The other results of dialogue management are the updating of mentioned resources, especially the dialogue history which can actually have a structure allowing it to link utterances to each other, and also objects and

the world in which the task is carried out, for example if the system has to modify one of these objects.

The goal of this module, at the heart of the system, is to make a decision which, on the one hand, will go in the direction of a natural dialogue with the user (we return here to natural dialogue in natural language) and, on the other hand, which manifests a perceptible persona to the user that is visible in the semantic content uttered and the method of its transmission. One of the system's roles is to be cooperative, and the modeling of this aspect is an especially important challenge, at least for closed domain systems such as the system covering train timetable information. We will thus explore the taking into account of natural and cooperating aspects in HMD (section 8.1) and then the technical aspects of dialogue management with a few approaches and challenges (section 8.2).

8.1. Natural and cooperative aspects of dialogue management

8.1.1. *Common goal and cooperation*

A natural dialogue is first defined as a discourse, that is a series of utterances creating a consistent whole which holds water. It is also defined as a finalized and collaborative activity (McTear, 2004). The first term highlights the goals which the speaker and hearer have in common, and in the case of a finalized dialogue, the task to be solved. This is the goal which will spur planning, that is the determining of plans, and thus utterances, to achieve a goal. When obstacles appear, it is because there is a missing element that prevents the hearer and speaker from achieving their goal. The second term highlights cooperation, that is the general principle stating the speaker and hearer are trying to help each other rather than be at odds and put the dialogue in peril. To have a dialogue, is to do everything to continue having a dialogue, as long as the goal has not been reached. This notion of cooperation which can at first seem rather abstract has been the focus of many research studies trying to deduce rules on which a dialogue manager could rely.

(Grice, 1975) considers that the dialogue is driven by a set of maxims, or major principles, which allow the hearer and speaker to interpret and generate relevant utterances. His *cooperative principle* is formulated thus: "Make your contribution such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged", and his maxims are as follows: maxim of quantity (make your contribution as informative as is required, do not make your contribution more informative than is required), maxim of quality (do not say what you believe to be false or that for which you lack adequate evidence), maxim of relevance (make your contributions relevant), maxim of manner (avoid obscurity, avoid ambiguity, be brief, be orderly). In spite of their vague aspects, for example for the maxim of relevance, these principles have had a determining influence on the

following studies. The Geneva model of discourse analysis (Roulet *et al.*, 1985) thus emphasizes two constraints arising from Gricean maxims, constraints which play a role in determining dialogue structure: interaction completeness, which tends to have the dialogue progress toward the satisfaction of both speaker and hearer, and interactive completeness, which is the tendency, when a conflict occurs, to solve this conflict. The Geneva model was then derived and extended, especially to integrate not only linguistic criteria (Reboul and Moeschler, 1998, p. 87), but also many other models, the starting point still remaining a cooperation principle followed by speaker and hearer. The maxim of relevance is completely reformulated by relevance theory which uses it as the guiding line of a very sophisticated approach on inferences carried out in cooperative dialogue (Sperber and Wilson, 1995). More recently, (Allwood *et al.*, 2000) described an approach of the cooperation mechanism principles in dialogue with joint purpose criteria, trust criteria and also cognitive and ethical consideration criteria, which broaden the spectrum of application of the Gricean maxims. The authors redefine the notions of coordination and cooperation versus these four criteria to variable degrees. The common goal (or joint purpose) is thus described as a degree of mutual contribution to a shared purpose, a degree of mutual awareness of this shared purpose, a degree of agreement made about purpose, a degree of dependence between purposes and a degree of antagonism involved in the purposes. Another example is the ethical consideration with, for example, the fact that we should not force others, or prevent them from following their own motivations.

All these criteria and all this work allows us to get a general idea of the coverage and complexity of mechanisms involved in dialogue. In addition, they lead us to conclude that implementing a cooperating HMD system is not an easy thing and that the links between the theories and the concrete examples are not easy to create. Some system designers redefine their notion of cooperation with preoccupations that are closer to technical aspects or directly implementable criteria. One of the aspects of cooperation in (Luzzati, 1995, p. 39) falls within the scope of the system's management of its own errors. Cooperation is materialized mostly by the choice of answers the system gives to the user's utterances. If we take as an example "how long with this itinerary?", the first answer that the system could give would be "two hours". In that case, the system, eventually referred to as a communicant, simply answers the question asked without adding to or subtracting anything from the queried value. A second possible answer could be "two hours due to a change at Versailles". In this case, the answer to the question is partly assessed and partly explained. It is assessed by the system which, based on criteria such as the average length of a single journey between Palaiseau and Paris, observes that a 2 hour journey is long and might not please the user. It is then explained by the system, which looks for and describes the main reason behind the length of time. With such an answer, the system can be considered as cooperating. However, it remains within its role of system in charge of satisfying the user's queries. Or, a third possible answer goes beyond this role: by answering "two hours, due to a change in Versailles, but if you go through Meudon you'll get there

in fifty minutes", the system shows an increased level of cooperation – we can then refer to it as a collaborating system – which is to suggest a change of direction to the user, trying a different itinerary than that the user had originally chosen. This change in direction, if the user accepts it, leads to the joint construction of a common goal.

8.1.2. *Speaking turns and interactive aspects*

The train ticket reservation task can lead to natural dialogues, as in the introduction's example, but also to situations which are closer to the caricature of a certain type of communication between a human being and a machine: "I would like to go to Paris", "what day?", "it will be tomorrow", "what time?", "let's say around nine o'clock", "what class?", "first, please", etc. The user can generate all kinds of imaginable utterance; the artificial aspect arises here from the system. By only generating questions and always phrasing them in the same way, the system is indeed helping the task progress but is not at all contributing to the linguistic aspects which make a natural dialogue in natural language, and we cannot consider its behavior as cooperating on an interaction point of view. A system may thus find it very beneficial to add interactivity, which can not only create variations in the types of phrase, but also vary the intervention content, and thus add, for example, a short utterance in charge of linking the user's intervention with the system's. (Denis, 2008, p. 67) thus provides various possible answers to the user's request "I would like to go to Paris":

– "when would you like to leave?": react on one of the missing parameters in the request, that is initiate a relevant contribution;

– "OK. When would you like to leave?": acknowledge and then react on the request;

– "to Paris. When would you like to leave?": repeat the only parameter given by the user, which allows it, on the one hand, to acknowledge it and, on the other hand, to let the user check that its request has been understood, at least on this point (if it is not the case, the user can then immediately react);

– "you want to go to Paris. When would you like to leave?": emphatic repetition, which can take on various aspects of paraphrase, from the user's entire utterance, which works as an acknowledgement as well as evidence that it is understood and allows the user to react immediately in case there is a mistake.

We remain here in cases in which the system speaks once the user has finished his utterance, and vice versa, in other words, in alternating interactions. Yet, conversational analysis studies have underlined the diversity, in human dialogue, of speaking turn organization phenomena, sequence and segment organization phenomena, reparation organization phenomena, etc. (Sacks *et al.*, 1974). The recorded corpora have shown that a dialogue is not just an ordered sequence of utterances. A dialogue between speaker and hearer involves two communication channels operating simultaneously: the main channel which is the speaker leading the dialogue at a given moment,

and the backchannel occupied, for example, by listening hints given by the hearer. These hints can be given as non-lexical sounds or short utterances with a speech act translating acknowledgement or even comprehension such as “hmm”, “yes” or even “oh really”. They can also take on the shape of completing the speaker’s utterance or even repeating part of it. In any case, because they are brief and do not really constitute a speaking turn, they do not prevent the speaker from going on with his intervention. This can thus lead to voices being superimposed. In HMD, some studies have integrated the generation of such utterances for the system’s behavior (Ward and Tsukahara, 2003; Edlund *et al.*, 2005), but there are many technical issues: speech recognition works at the same time as the system is making noise, which can lead to lower performances; a control utterance generation can happen at the moment when the user was about to stop talking, which can create a small moment of uncertainty and, in general, generating a control utterance is not always done at the most relevant moment, due, for example, to the slight temporal delay with regard to the best places, the TRP (see paragraph 2.2.1). However, it is true that giving the system the ability to use the backchannel increases the realistic aspect of the dialogue, providing it with the ability to occupy the field when the user is not saying anything, that is to generate a reviving message when the user is not answering, to generate a dialogue maintenance message when the user does not know what to say and even to generate a waiting message when a process may take more than a few seconds.

8.1.3. *Interpretation and inferences*

To react in a completely relevant manner to an utterance, understanding is obviously necessary as well as understanding its implicitations and explicitations (see section 5.3). Determining the semantic content and speech acts is the first step, but it is also necessary to make the right inferences, which are based on the utterance and its context and allow the human hearer to understand the allusions and other implicit contents immediately. In this case, (Grice, 1975) also suggested a terminology and general principles which allows us to better understand the dialogic phenomena. Using the term of implicature for pragmatic inferences, he draws a distinction between conventional implicatures which are triggered by a conventional use of language, conversational implicatures, which are triggered by the utterance’s link in the context of its enunciation, and generalized conversational implicatures, which are triggered in a contextual manner without any help from the utterance’s linguistic elements. These types of implicatures allow us to explain understanding and cooperation in a dialogue. It is when the speaker seems to violate one of the maxims that we have to look to implicatures. Calculating an implicature is to determine what the speaker implicitly supposed, to keep the cooperation principle intact (Denis, 2008, p. 18).

Moreover, a dialogue is not only an exchange of information whether it is explicit or implicit. It can also be a negotiation trying to convince the hearer or to prove something. This aspect is not very prominent in HMD, but is the focus of work which

leads to the identification of speech acts which are is some way enriched by being put into perspective. Among the three types which we have seen in Chapter 7, the assertion is, for example, not very explicit with its underpinning intention. (Baker, 2004) emphasizes that in a negotiation dialogue, it is not about creating assertions as much as making proposals or offers. An assertion can provide an argument, and this is also one of the aspects of dialogue which is not greatly taken into account by HMD systems.

8.1.4. *Dialogue, argumentation and coherence*

Linguistics and pragmatics researchers such as J. Moeschler have long studied the argumentative dialogue and drawn links with speech act theory and even relevance theory. They thus define a specific category of speech acts, the *argumentation acts*, that are carried out when an utterance is meant to serve as a conclusion (Moeschler, 1985, p. 189). The linguistic form itself being able to contain argumentative instructions, for example argumentative connectors "but", "however", "thus", "because" or "so", the interpretation of an utterance gives it an argumentative dimension which consists of identifying the (semantic) direction and type of (pragmatic) act.

As soon as several utterances are involved, it is also about linking them to each other through argumentative relations which allows us to structure the dialogue by detecting how an utterance can provide an argument in the direction of such an assertion. All this gives an analysis dimension which we have not yet mentioned and which consists of adding a set of additional labels to each utterance, containing an indication on the act and the utterance's direction and on its relations with previous and subsequent utterances. It is actually here that we can observe that the implementations in HMD are very few: this analysis dimension, a bit like that of the Geneva hierarchical model of dialogue analysis, can mostly be applied afterwards on a corpus rather than on an effective system in real time. Moreover, our ticket reservation examples do not really involve an argumentative dimension, and it is mostly a challenge for the recreational or open domain HMD systems.

Argumentative dialogue management is accurately described from a theoretical point of view, especially by (Moeschler, 1985), who suggests a set of dialogue strategies such as: anticipated negotiation, an argumentative strategy aiming to anticipate any counter-arguments that could be opposed and immediately refute them ("you could believe that it is more expensive by going through Versailles, but it isn't…"); factual negotiation, a discourse strategy which focuses on agreeing on the basis of certain decisive facts to carry on the interaction; interaction negotiation, a strategy aiming to impose a image of oneself and the other during the interaction; meta-discursive negotiation, aiming to give indications which allow one to retroactively interpret the function of an intervention or even meta-interaction negotiation, aiming to define the rights and obligations of the speaker and hearer. To correctly apply these strategies,

we have to identify the chains of discursive acts, which J. Moeschler suggests thanks to the notion of discursive movement, a kind of structure that brings together several argumentative acts, and on the basis of their direction allows him to identify the cases of argumentative concession or conclusion.

Moreover, managing an argumentative dialogue involves interruption possibilities: contrary to a request or narration, for which the hearer usually waits for the end of the utterance, an argumentative intervention can spur the hearer to interrupt the speaker, to kill an argument before it is completely uttered. This is what (Dessalles, 2008, p. 17) shows by starting from the observation – which he arrived at after a corpus study – that hearers are often able to anticipate the nature of the argument before it is completely expressed. The interruption then becomes a (somewhat brutal) dialogue strategy which can be taken into account by an HMD system. Even though these dialogue strategies can theoretically be applied to any type of dialogue, let us, however, note that they remain more relevant in the case of negotiation and argumentation dialogue than in the case of command or information dialogues.

A final aspect that also takes on a particular importance in this type of dialogue is coherence and cohesion management. We have to identify the relations between several utterances, a bit like the argumentative relations do. (Prévot, 2004) summarizes this question which has sparked many linguistic research studies, and draws a line between a semantic coherence, which is mostly spatial and temporal; an implicit coherence, covering the aspects related to intentions, content and some dialogue conventions and surface hints which define cohesion: information structure, ellipses, anaphora and coreference chains, that is linguistic phenomena which go beyond the borders of the sentence. (Moeschler, 1985, p. 190) adds argumentative coherence, which characterizes a discourse or dialogue in which the instruction given by the argumentative connectors is satisfied and any argumentative contradiction is solved. All these aspects allow an HMD system to knowingly manage the dialogue, that is to use a coherence and cohesion indicator and have additional indications for its choice of answer, a process which we will now investigate.

8.1.5. *Choosing an answer*

When the user asks the system a question, the choice of answer is imposed on it: either it knows the answer and gives it or it does not know the answer and apologizes, tries to redirect the user on another path or, potentially, asks the user to rephrase his question. In the context of a dialogue built over several interventions, a question can involve much more complexity. (Luzzati, 1995, p. 61) shows us that in the case of train ticket sales, it is common for a human teller and thus a system to generate *maximalist* answers, that is answers which systematically have more information than is required for reasons of efficiency, especially to prevent potential follow-up questions. This returns slightly to the emphatic repetition principle mentioned in paragraph 8.1.2.

Moreover, to better determine when the system can generate this type of answer, which violates one of the Gricean maxims of quantity, it is helpful to characterize the different types of possible question. (Van Schooten *et al.*, 2007) studied the importance of follow-up questions and were led to suggest a question typology. Among the types of question, we have already seen examples of closed questions, for which the possible answers are "yes", "no" and "I don't know", and examples of open questions, for which the answer is a propositional value such as a referring expression, a quantity or a named entity, but we have not yet met questions such as "how can I make a reservation?". However, this question requires an explanation, which can be hard for an HMD to carry out.

When the user gives the system an order, the choice of answers is also imposed on it: at the same time as the system carries out the order, at least if all the conditions are fulfilled (see paragraph 6.2.1), it can generate an utterance which on the one hand announces the action being carried out (especially if it is invisible) and on the other hand allows the user to follow-up and continue the natural dialogue. Following the order "reserve a first-class ticket for Paris tomorrow morning", a system's utterances such as "OK" or "done" are probably not sufficient, because it closes the dialogue. An utterance including a revival is more relevant, such as "it's done, would you like another journey?" and does not put the user in a position in which he does not know what to say.

When the user generates an assertion, such as, for example, "I do not have a senior discount card", this is the point where the system must prove it is able to manage a dialogue: an assertion contributes a seemingly new piece of information to the system or it is useless, and this newness must trigger inferences. On the basis of these inferences, and on what has already been said and what both speaker and hearer know, the system should understand if the assertion is filling a lack which was blocking a situation, which the inferences unblock and will allows the system to know what to answer, or, on the contrary, if the assertion should incite the system to suggest something such as "let us see if you can get one. How old are you?".

Moreover, a user's utterance, no matter its speech act, can cause the system to react in an unexpected manner, for example when it is unable to solve an ambiguity on a referent or simply unable to understand the utterance. In this case again, there are various dialogue strategies available. (Denis, 2008, p. 43) is particularly interested in detecting issues in system robustness and comes to suggest dialogue strategies in the cases where there are compound issues. Thus, a classical vision of ambiguity management consists of choosing among the alternatives, even if the wrong choice is made, rather than commit to a clarification sub-dialogue which might give a negative image of the system and also lower the chances of quickly satisfying the task. Making up an error on a referent can indeed end up being quicker than a clarification sub-dialogue. A. Denis adds a phenomenon to this vision that was observed in corpus and has not been much studied in HMD, i.e. the case in which the clarification request itself

is source of incomprehension or of a divergence of interpretation between user and system, which can lead to an inextricable situation. The existence of this possibility reinforces the relevance of the strategy which is to force the system to choose.

8.2. Technical aspects of dialogue management

8.2.1. *Dialogue management and control*

It is not easy to breakdown the dialogue management process into tasks. Many approaches have come one after the other, and it is hard to place them in relation to each other given the diversity of settings used and the overlap between processes. In general, *dialogue management* groups three phases that more or less overlap one another: first, dialogue control which tries to manage the interactive process to determine a type of reaction after a structured chain of utterances, and second, the modeling of the dialogue context, which focuses on the dialogue history and the way in which the utterance content is grounded, and finally the dialogue initiative which adds a specific behavior to the previous considerations (McTear, 2004; Jokinen and McTear, 2010).

Dialogue control is the focus of this section and has been the focus of a great many studies (Pierrel, 1987; Sabah, 1989; Carberry, 1990; Luzzati, 1995; Traum and Larsson, 2003, etc.). If we exaggerate slightly, firstly there are the methods of finite states that have been implemented: they are finite state or *dialogue grammar* automata, and the point is to determine all the possible situations and ways of going from one situation to the next. This type of approach works very well for dialogues in which the initiative is always coming from one side, for example the system's side, because in that case they are the system's utterances (questions in very directing systems, different speech acts for more flexible systems) which are taken into account by the states, the user's answers being taken into account by the transitions (Jurafsky and Martin, 2009, p. 863). Then come the (fill in the blanks) pattern methods, which allow us to recognize situations without forcing the initiative to remain on one side or the other.

Then come the methods based on an information state, that is the methods which add a memory, which contains anything required, knowing that this memory will help determine the possible follow-ups of the dialogue. Depending on the authors, the information state contains the dialogue history, common ground, a model with mental states, a model of the user, etc. The important bit is to use and update the data which will behave a bit like the global variables in the dialogue manager. Thus, the planning approach is very successful in initial studies by (Cohen and Perrault, 1979) and (Allen and Perrault, 1980): the goal is to recognize and plan the plans, speech acts being planned just like actions (the speaker's speech acts are part of a plan that the hearer must unveil to answer in a relevant manner), based on a modeling of the speaker and hearer's mental states. To reconcile natural dialogue and task satisfaction, we can distinguish two types of plan managed in parallel: the discourse plans and the

domain plans. In general, the plans allow for a great number of possibilities in terms of dialogue control, see the article by N. Maudet in (Gardent and Pierrel, 2002).

Then come the joint action theories (see paragraph 8.2.2 with the notion of common ground) and dialogue control inspired by game theory, each utterance being a turn played during which the speaker tries to maximize his gain (Caelen and Xuereb, 2007). Finally, machine learning techniques have appeared more recently for dialogue control with Markov decision process (MDP) and partially observable MDP (POMDP) model types, which extend that of the information state by adding a probabilistic means to decide on the future action depending on the current state (Jurafsky and Martin, 2009, p. 883). In the same vein, (Singh *et al.*, 2002) use reinforcement learning to achieve an optimal decision set. The decision rules are refined by carrying out several thousand of exchanges between the system and a simulated user.

As we can see, dialogue control can involve many kinds of techniques. Let us add that implementing them can involve other data in parallel, such as an analysis of the topics approached, for example, so as to direct the system's reaction toward one of the current topics rather than one of the abandoned topics. This is the approach given by (Vilnat, 2005), who has three distinct cooperating pragmatic analysis submodules:

– a *thematic interpretation* module managing the global coherence of the topics approached during the dialogue;

– an *intentional analysis* module which provides functional dialogue representation in which the roles of the various interventions are clarified; see the presentation on the approaches based on intentional structure in the article by N. Maudet in (Gardent and Pierrel, 2002);

– an *interaction management* module which allows the system to react to different types of incomprehension by allowing the dialogue to remain efficient.

As for (Rosset, 2008), she adds strategies depending on ergonomic choices and suggests a dialogue model built around a set of phases: acquisition (obtaining the information required to satisfy the task), negotiation, navigation, post-acknowledgment (transition toward a negotiation, navigation, or end of dialogue) and metaprocessing (marking and processing errors); see the ARISE system (Lamel *et al.*, 2003).

8.2.2. *Dialogue history modeling*

Modeling the context of the dialogue can also lead to a great variety of forms in theoretical models and implemented systems. The first aspect consists of providing contextual elements to completely understand an utterance, after the semantic and pragmatic analyses have been carried out, that is at the level of the role of the utterance in the dialogue's progression and, for example, in satisfying the task. Among these elements we can thus find hypotheses on the user's beliefs or desires, which can help

put the semantic content of his utterance into perspective. We also find all kinds of information using what has already been said during the dialogue, and thus calling upon the history. This is the second aspect of dialogue context modeling: formalizing and storing in a data structure the contextual interpretation results as well as highlighting in this history a structure describing the dialogue's progression, for example the path followed to solve the task.

A notion that has emerged and generated many proposals in theoretical work on dialogue is that of *common ground*, that is the information set shared by the speaker and hearer, either because it mutually manifests, for example because it has been verbalized, or because it can be deduced from what has been said: for example, when the system answer "to Paris. When would you like to leave?", it is obvious for both the speaker and the hearer that the user wants to go to Paris and the system is aware of this desire. The common ground becomes what the speaker and hearer build as the dialogue progresses. Therefore language is seen as a *joint action* (Jurafsky and Martin, 2009). More precisely, we can distinguish between a communal common ground, where the shared knowledge goes beyond the interaction between the speaker and the hearer, and a more personal common ground in which the knowledge is only good for the speaker and the hearer (Denis, 2008, p. 45). To be part of the common ground, a speaker's utterance must be grounded by the hearer. We thus distinguish between the *grounding process*, a process during which the speaker and hearer update the common ground, and the *grounding criterion*, a criterion which the speaker and hearer are aiming for, with a desire for a joint belief of understanding. Several grounding models have been suggested. The first models, which do not draw any of these distinctions, automatically update the common ground at each utterance. As a reaction, (Clark and Schaefer, 1989) suggest the *discourse contributions* model, in which the grounding can only happen when the grounding criterion is reached. Other proposals try to go down this path to reach implementable models, especially the grounding acts model with its nine levels of grounding, from "unknown" and "not understood" to "accepted" (Traum and Hinkelman, 1992), then the weak belief model which provides an explicit modeling of understanding beliefs necessary for grounding: it considers that the speakers and hearers carry out hypotheses on the understanding of their partner and that the confirmation of these hypotheses allow them to reach the grounding criterion. A reinforcement mechanism then helps transform a mutual weak belief into a mutual belief; see (Denis, 2008). Finally, a more recent tendency consists of integrating into grounding models numerical aspects with, for example, the calculation of a score which allows the system to characterize the point of grounding a piece of information. As is often the case, this numerical approach complements symbolic approaches.

If we now take up our favorite example and the distinction of three main speech acts from relevance theory, the reception by the system of the utterance "I would like to go to Paris" leads to the set of following reasonings: analysis of the utterance's meaning which leads to the identification of the expression of intent. The system thus

remembers this intent and this will allow it to plan its next actions. The system's roles are to inform the user, here on the ways to reach Paris, and sell him train tickets. It is highly probable that the utterance of "I would like to go to Paris" by the user is an expression of a desire, that the system help him reach Paris. Satisfying this desire means identifying, among the set of possible ways to reach Paris, that which will best satisfy the user. For now, the system does not know his preferences, however, it can always sort the itineraries from its database to suggest the most relevant journeys.

At this point, the system thus decides to suggest something to the user and includes several alternatives so as to give him a choice. The system also decides to manifest an acknowledgment and use multimodality, that is to display several itineraries leading to Paris on the screen (and highlight the Paris train stations involved) at the same time as it utters a relatively short sentence, "here are the possible itineraries", a sentence that does not include the information on the trains which is too long to verbalize. Thus, the utterance's grounding by the user is carried out by highlighting the Paris train stations, and the acknowledgment of the user's mental states is shown in the oral utterance. We can consider this answer to be cooperative: it is neither too short nor too long, it is relevant given the user's utterance, involves the recognition of an intention and an indirect speech act and it globally satisfies the theoretical criteria described in the previous sections. As well as generating this answer, the system stores in its modeling of the dialogue history the identified mental states as well as this first materialization of a plan consisting of identifying a way to get to Paris to sell the matching train ticket.

The user's following utterance, "how long with this itinerary which seems shorter?" creates all the issues which we have described in the chapters in Part 2 of this book, and gives rise here to new questions: what does this utterance contribute on the mental states and started plan? By interpreting the utterance, the system first realizes that its intervention has been understood: the user has indeed seen the suggested alternatives because he is asking a question on one of them. There is thus a grounding of the system's answer. Moreover, the question leads to the identification in the applicative database of the property of an itinerary, and the assessment of the comment leads the system to carry out a comparison based on the itineraries' properties. This assessment is triggered because the system has decided that the comment was actually a question through an indirect speech act. If this indirect question is relevant, it has to answer it, and thus needs to compare the lengths. After this comparison, the system knows that yes, the itinerary the user pointed at is indeed shorter than the other displayed itineraries, that is the shortest solution. It can thus answer the indirect question. As it has been expressed as a comment, the system manages in parallel the belief linked to it, that is the user believes that this itinerary is probably the shortest. At this point, the user's mental states have been updated as has the dialogue history, and the system is faced with several facts: it knows the answer to the direct question, it knows the answer to the indirect question and it is still trying to satisfy the user by inciting him to choose, quickly if possible. This can lead the system, in the example which appears in the introduction, to generate the answer "twenty minutes". The answer is rather short,

but matches the system's decision to answer the first act quickly and efficiently and to ignore the second act, the comment. This choice was made for two reasons: on the one hand because the system believes that the answer "twenty minutes" will satisfy the user (it is a short and thus satisfactory length of time), on the other hand because it considers that confirming the belief, in this case true, is not essential. Finally, in addition to the generation of an answer, the system updated the dialogue history with a description of everything that we have just seen and the materialization of the progress of the dialogue, especially the fact that the system's answer continues to focus the dialogue only on the itinerary which the user pointed at. Here we find certain aspects that we had already discussed in Chapter 5.

To implement this type of process, we see that linguistic and pragmatic analyses are essential, with, for example, the identification of indirect speech acts, that the identification of the user's mental states contributes reasoning possibilities and the management of a correctly structured dialogue history is also indispensable. As an example, (Vilnat, 2005) suggests the notion of a history page that contains the speaker's identifier for each intervention, the semantic and pragmatic representations, the topic concerned, the goal concerned, the state of the dialogue's structure, the state of the interaction variables, the state of the plan being developed, etc. As we can see, the history page is a complex, multiform and multifunctional structure.

If we take up the three main speech acts, we can consider the following processes and structures:

– "saying that": the speaker expresses an assertion in order to let the system know something. The system updates its database which is part of the common ground and can be referred to as CG. Indeed, by providing the system with information, the speaker helps to make this information mutually known and manifest (at least after taking the grounding process into account);

– "telling to": the speaker gives an order so that the system does something. It updates a list of actions to be carried out, which we can refer to as a *to do list* (TDL). It is a sort of pile (or heap) cataloguing the things to do as the dialogue progresses and taking an item out as soon as the matching action has been carried out. Managing this type of list allows the system to know what it still has to do without it being linked to the current utterance's processing but on the contrary, making it possible to carry out an action several speaking turns after the user's request;

– "asking": the speaker expresses a question in order to learn something from the system. It updates the list of questions which it has to answer, which we can call *questions under discussion* or *questions under debate* (QUD), that is a similar structure to the TDL, in charge of cataloguing and managing the questions asked during the dialogue.

Depending on the approaches, only one of the three structures can be updated during a speech act's processing or, on the contrary, multiple updates are allowed. In

his successive models, (Ginzburg, 2012) detailed the operations of substructures and suggested complementary structures, for example *latest move* (LM), *shared ground* (SG), *facts* and *pending* (transient structure). Other approaches highlight a structure cataloging the speaker's commitments, *commitment store* (CS), a structure specific to salient utterance, *salient utterances* (SAL-UTT), or even a subset of the QUD structure in charge of a specific type of question, or *issues*, following a distinction between various types of question depending on their function in the dialogue (Denis, 2008).

8.2.3. *Dialogue management and multimodality management*

The previous example had a dialogue management initiative (see paragraph 8.2.1): when the system decides to answer "here are the possible itineraries" and displays a set of alternatives on the visual scene, it makes a choice about what it is about to say and display. The display of possible itineraries and the direction to Paris fall under the scope of an act of a "saying that" type, except it is "saying visually" instead of a "saying". The utterance of "here are the possible itineraries" falls under the scope of a "saying that". The system could have made other choices, such as asking a question like "would an itinerary with a change in Meudon be acceptable?" which would advance the task and test one of the alternatives. The utterance would then be characterized by the speech act "asking". The speech act choice in reaction to the user act is thus part of dialogue management. In the multimodal context that we are studying here, the chosen act is a multimodal composite act: the system wants to indicate various possible choices to the user and does it in a certain way that strongly involves the functionalities of multimedia information presentation.

Dialogue management has other aspects that our example does not give, for example clarifying the conditions that allow the system to abandon its goals, decide on a path aiming to satisfy a specific goal and in general explore the interaction between goals, beliefs and intentions. This is what (Cohen and Levesque, 1990) show with the example of a robot that states it is going to bring something, does not, and then explains that it found something else to do, and this example illustrates rational action theory. Moreover, dialogue management also includes the ability to manage incomprehension (see paragraph 8.1.5 with the same type of preoccupations for ambiguity management). The system can choose to solve the incomprehension without the user (we then refer to internal robustness) or to solve the incomprehension with the user, that is by starting a clarification sub-dialogue (external robustness). As (Denis, 2008, p. 35) shows us, the two approaches are complementary: we cannot make do with a very good internal robustness and a weak external robustness, as the dialogue also has a function of talking about what is not going well, and some issues require a clarification request. However, the opposite is also unacceptable: a system systematically relying on the user to solve its interpretation problems ends up being annoying. This is the case, for example, of some of the first systems with online machine learning,

which in order to check they had integrated a new term properly or to see they had understood correctly, asked a closed question at almost every speaking turn.

Multimodal dialogue management and, for example, dialogue management for an information system, that is a system devoted to presenting complex information such as geographical data, have many other aspects specific to the management of this amount of information. If the system decides to present the details of 30 train itineraries, or to show a geographical map annotated with the elements answering the user's query, it must be able to control the way in which it transmits this information. This can be done by planning, that is allocating the transmission over several speaking turns. This can also happen by allocating the information over various communication possibilities, a bit like "here are the possible itineraries", but mostly as we will see in Chapter 9. On this criterion of information quantity close to the notion of cognitive load, (Horchani, 2007) presents and models three dialogue strategies following a query such as "I would like to go to Paris":

– Enumeration: in the case in which the number of solutions is reasonable (but then a threshold has to be decided upon, especially since this threshold depends on the average amount of information contained in a solution), the system presents a list covering all the solutions, and this list can be verbalized, displayed, or partially verbalized and partially displayed.

– Restriction: in the case in which the number of solutions goes beyond the reasonable threshold, the system suggests criteria to limit the scope of the research space. The system can also suggest conditional answers. More than the transmission of an answer, the dialogue act here is the transmission of conditions to solve a problem.

– Relaxation: in the case in which there is no solution found, the system suggests either alternative solutions or alternative research criteria. The potentially displayed answers are suggestions rather than answers.

In a multimodal context, dialogue management also has the temporal management of speaking turns, especially when research and, even more, the presentation of complex information take time. We have seen that it was not necessarily relevant for the system to interrupt the user. However, the question can arise differently when it is not trying to create an oral utterance but start a multimedia information presentation. Indeed, a purely visual action from the system's part can be considered even when the user is talking, especially if this visual action can provide quick information which would be efficient for the interaction's progression. Beyond the aspects linked to multimodal message generation, which we will explore in Chapter 9, these are indeed aspects linked to dialogue management, which are at stake here.

8.2.4. *Can a dialogue system lie?*

One last aspect on which we have not insisted enough is the importance of the task in dialogue management. The introduction's example will allow us to illustrate several system behaviors depending on its own priorities, and especially the ability to lie, a fascinating behavior for an HMD system.

Let us return to the dialogue management at the moment of the U2 reception, "how long with this itinerary which seems shorter?". In paragraph 8.2.2, we suggested that the journey length was 20 minutes, which the system assesses as short, and we supposed that the comment was true, that is the designated itinerary was the shortest among the identified solutions. Let us now suppose, as in paragraph 8.1.1, that the journey length is 2 hours, which the system can assess to be a long time, which might not be satisfactory given the matching distance. Let us add to the system's priorities that of selling a train ticket, a priority that can be materialized in various ways: have the dialogue end on a sale and not simply a timetable information request, suggest promotions, prioritize certain ticket sales over others depending on constraints such as the very close travel date, the length of ticket validity. In the end, the system can be made to generate the utterance "two hours" with no more precision for one of the following reasons:

– The system knows that it is an unsatisfying length of time, but it is the shortest journey – which means the user's comment is true – and it does seem to be the best solution. The system does not consider that it is necessary to confirm the comment, for reasons mentioned in paragraph 8.2.2. It may also not have dared to tell the user to take a taxi, maybe because it falls outside of its field of expertise.

– The system sees that it is not the shortest journey, but the other possible itinerary lasts ten minutes less and involves an additional change. Overall, the user's belief is false but this is not really a problem. Instead of losing time explaining the pros and cons of each possible itinerary, the system chooses to answer the direct question and ignore the comment.

– The system sees that it is not the shortest itinerary and that there is another possibility that takes just under an hour. Based on pragmatic priorities, it considers, however, that answering the first act is much more important than answering the second, and that it must favor generating a short answer above all. It will thus answer with the strict minimum. If the user is not satisfied, he will always ask a question about the other itinerary displayed.

– The system sees that the other possibility is indeed faster but also much cheaper: prioritizing the sale of a more profitable ticket, it decides not to say anything, hoping that the user will not ask any additional questions and remain focused on this itinerary.

In either of these cases, answers such as "two hours, and it is the shortest itinerary", "two hours, the other itinerary is slightly shorter" or even "two hours because of a

change at Versailles" should be preferred. There is no visible lie, however, unless by omission. It might seem hard to consider that a system can deliberately lie, by giving false information, for example by answering "one hour" when the journey lasts 2 hours: on the one hand, the user might notice it and on the other hand, it requires a duplication of the application's database in order to avoid any subsequent contradiction. We can hope that no system designer ends up having to go to such ends.

To end on this example, let us note that there is a close relationship between composite speech act management and dialogue management: in a case in which the length is 2 hours and it is not the shortest itinerary, the system is faced with several reaction possibilities, depending on the priorities given to the two speech acts and the way to make the dialogue progress:

– When a user expresses a belief and this belief is false, the system can consider it a priority to re-establish the truth. It can then overturn the linguistic importance given to the first act then to the second, and decide to react only to the second act ("no, it is not the shortest itinerary") or to both acts, but by starting the answer with the second act ("no, it is not the shortest: it takes two hours"). In the case of a cooperating system, it can even add the transmission of the shortest itinerary's identifier, that is of the solution that makes the comment true: "no, it is not the shortest. Here is the shortest itinerary", highlighting the matching itinerary on the screen.

– No matter the importance given to a comment being true or false, the system may consider that from a linguistic point of view it must first answer the first act and then the second act, which gives "two hours, but it is not the shortest itinerary" or, with a cooperating behavior, which is greatly appreciated, "two hours, but the shortest itinerary is this one".

Many other answers are possible, and this illustration shows on the one hand that generating answers in natural language to increase the HMD's realism creates many issues, and on the other hand that a fine identification of semantic content, speech acts and the hearers' mental states is required for the system to achieve an adequate behavior, i.e. behavior that is comprehensive, relevant, coherent and adapted to the task being solved.

8.3. Conclusion

The task to be accomplished is the driving force behind the dialogue: the dialogue progresses when the task progresses. However, a realistic human-machine dialogue should not be built around this priority alone: it also has to take into account fluidity and the linguistic spontaneity of exchanges. This chapter confronts behavior versus the task and the linguistic behavior, to show how to get closer to a natural dialogue in natural language. Dialogue strategy examples illustrate not only how a system can be optimized in that direction but also how a system can be made to lie.

Chapter 9

Multimodal output management

Each time the dialogue manager decides to generate a message for the user, which usually happens a short time after the end of the user's intervention (but can also happen in the middle of an utterance), an automatic generation process is launched. For written or oral dialogue, it is the domain of text generation that is concerned. For multimodal dialogue, whether it is an information system able to display complex data, a system managing a microworld represented on a screen, a system with a force-feedback device, an ECA or a robot able to generate gestures when talking, the generation of a natural language utterance goes hand in hand with that of a gesture or a visual return. The process can then involve multimodal generation, that is the generation of multimodal references in the opposite direction as that we have studied in Chapter 6, as well as the transmission of multimedia information. For this last point, the field concerned is that of multimedia information presentation systems, called IMMPS (Intelligent Multi-Media Presentation Systems) (Stock and Zancanaro, 2005), a full-fledged field of research similar to that of ECA. An HMD system's output management can involve many processes spread over multiple modules.

To approach these processes, we can draw a distinction between the *what* and the *how*. The first falls within the scope of the dialogue manager (Jurafsky and Martin, 2009). It integrates a *what to say* and potentially a *what to display* and a *what to do*, each of them including semantic content and, especially in the first case, a dialogue act. The second falls within the scope of text generation and is the one we will study in this chapter. These are the successive processing steps for text generation: content planning, i.e. choosing the way to dispose the different propositions constituting the semantic content; sentence aggregation, i.e. allocating propositions to sentences and determining the discourse relations; lexicalization, that is the choice of words; referring expression generation, for now only in a linguistic framework, which requires the

system to choose between direct reference and anaphora; linguistic realization, with the application of syntactic and morphological rules to obtain a well-built sentence (Reiter and Dale, 2000). Within the framework of oral dialogue, we can add a prosody determination phase, and a text-to-speech synthesis phase, which can include an oral rendition of emotions and dialogue act management, notably with the generation of an act that materializes the change in speaking turn. If the dialogue involves an ECA, the *how* can also be broken down into several processes: choosing a type of physically perceptible behavior, taking the *what* into account, then rendering of this behavior, see Chapter 9 of (Garbay and Kayser, 2011). The management of a speaking head specifically requires a face animation phase (lips, eyes, hands, body, in general) including the visual rendition of emotions. All these processes involve various techniques, from the use of patterns, whether they are syntactic, prosodic, gestures, animations, with or without variables that can be set, to the management of linguistic and discourse phenomena as is the case for natural language generation when it uses information structure principles.

The links between the system's general behavior and all these processes is sometimes hard to draw. The rendition of emotions is a favored means to transmit a few indications, for example on the positive or negative direction of the message. Making this direction vary depending on the answers, the incomprehensions, the ambiguities or simply the system's inability to answer a request increases the realistic aspect of the interaction: a systematically positive interaction can irritate the user in case of repeated incomprehensions and, obviously, a mainly negative direction does not help with the cooperating aspect of the dialogue. Beyond the simple positive or negative direction (some systems can get upset when they detect malicious behavior on the user's part), the current emotion models involve various dimensions, each one potentially able to materialize itself on several modalities: valency (positive or negative), activation (weak to strong), the level of control (fear is not, for example, linked to a feeling of situation control, whereas anger is), and the level of unexpected; see Chapter 3 of (Garbay and Kayser, 2011).

To achieve a satisfactory level of realism, the system's behavior can also integrate certain aspects of spontaneous communication described in Chapters 3 and 5. It can, for example, generate hesitations and repetitions like a human speaker would (Rosset, 2008, p. 84), or even manage the cognitive load of the user, as we will see with human factors that are the focus of this chapter, first with a few general principles to design modules in charge of the system's output (section 9.1), especially at the pragmatic level of dialogue acts (section 9.2), then the description of several important processes in multimodal dialogue (section 9.3).

9.1. Output management methodology

9.1.1. *General principles of output multimodality*

A multimedia information presenter is meant to translate messages coming from the dialogue manager taking into account, in the best possible way, the specific characteristics of the information to be presented (and thus displayed or verbalized), of the terminal on which the dialogue is carried out, the physical environment (dialogue in a noisy environment, on an airplane, on a field of operation) and the user. When the information is to be allocated to several communication modalities, we speak of *multimodal fission*, the process with an opposite goal from that of multimodal fusion described in Chapter 6. The term *information* covers natural language or multimodal utterances as well as data stemming from the application model such as the characteristics of a set of trains. This information can be given labels describing its status when taking the task in progress into account: urgency or importance aspect (e.g. critical). Other characteristics can also be labeled or calculated by the presenter to test the presentation possibilities: discrete or continuous aspect, volume, complexity, number of elements (paragraph 9.1.2). This, in particular, allows management of totally different natural language utterances and data such as geographical maps and timetable databases. Rather simplistically, the dialogue manager decides:

– *who*: to whom the information has to be presented;

– *what*: what is the information to present;

– *which*: which part of the information has to be emphasized;

– *where*: where can the information be displayed, that is on which devices;

– *when*: when and for how long must the information be presented.

The multimedia presenter implements these decisions, that is it decides on the *how*. This is carried out by choosing the device(s) to be used, allocating the information to determine the part given to each device, dividing it to allocate the presentation during the imparted time, choosing the way in which to highlight the part in question, potentially managing a display-specific interface, with, for example, graphical metaphors such as sliders and navigation buttons in the space occupied by the information.

We can summarize the preoccupations of a multimedia presenter in a set of general principles such as the Gricean maxims. The design of systems including a multimedia presenter requires finely taking into account pragmatic and cognitive aspects of communication, and this is why these principles are uttered, and still have to be materialized (such as the Gricean maxims) by a theory such as relevance theory (Sperber and Wilson, 1995). A first aspect concerns taking into account the information characteristics and their anchoring in the dialogue history, which involves, in the case of communication including HMD and HMI, the interaction history storing all the direct manipulations carried out on the HMI objects. The first principles for natural, adaptive and user-centric multimedia presenters design are as follows:

– present well by allocating the information to the communication channels in a relevant manner;

– present well by taking into account the rendition and valorization of the information on each communication channel;

– present well by using the message's semantic content in a relevant manner;

– present well by maintaining coherence and cohesion with the previous messages.

A second aspect covers taking into account the terminal and the physical and situational environment's characteristics:

– present well by using the presentation means in a relevant manner;

– present well by using the presentation conditions in a relevant manner.

We then arrive at the phase of taking the user into account, with his physical and cognitive abilities, his roles in the task in progress, and his communication preferences as they have been defined and identified during the interaction:

– present well with a more refined use of the user's expectations;

– present well to prioritize an adequate perception of the message;

– present well to prioritize the adequate reactions on the user's part.

9.1.2. *Human factors for multimedia presentation*

Adapting to the user's physical and cognitive abilities falls within the scope of human factors (see the beginning of section 2.1). This is the field of cognitive psychology and especially cognitive ergonomics (Gaonac'h, 2006), with preoccupations such as that of cognitive load management, user memory and attention span and preoccupations that can be usefully applied to HMD, see Chapter 9 of (Cohen *et al.*, 2004). This adaptation completes that explored by plastic HMI (paragraph 2.3.2), with various aspects characterizing the HMI adaptability and more generally that of HMD, to terminals, user rights (right of access to certain pieces of information and not others), the roles of the user (depending on his role in task resolution, some pieces of information are more important than others) and preferences: preferences on data filtering, on modalities on be prioritized or on the ways to highlight part of the information. All these processes intervene in the allocation of information to communication channels and the highlighting of specific pieces of information.

To carry these processes out, a first set of parameters covers the characteristics of the information to be transmitted, with three main categories: semantic content, pragmatic aspects and anchoring in the dialogue history. Among all these elements, which create the semantic content, we will note the following essential parameters:

– the level of criticality, which can lead to a choice to emphasize it greatly;

– the level of urgency, which can block all the current processes to force the user to react immediately;

– the complexity of the information: nature of the data structure, volume and number of elements involved;

– the constitution of information: discrete or continuous, linguistic, numbered, spread over one, two or three dimensions;

– the extent of the information, with, for example, a choice consisting of showing all of the information, which might make it impossible to read, and add a zoom on a focalized, and thus legible, part;

– the presentation constraints inherent to the multimedia information itself: visual for cartography, without constraints for a message in natural language, which can be displayed or verbalized.

The pragmatic aspects cover the illocutionary value and force discussed in paragraph 7.1.1, that is the dialogue act carried by the action of presenting, with several levels depending on the marked or unmarked aspect of this act, as well as the perlocutionary value and force, which reflect the effects carried out by the expression of this dialogue act and that we will see in section 9.2. As we saw with input processing and dialogue management, these pragmatic aspects take on a meaning when they are linked to the dialogue structure, and thus to the modeling of the multimodal history: a history of the messages exchanged, of the data displayed, the display actions carried out, to authorize subsequent mentions of these actions.

A second set of parameters covers the presentations means and conditions. The presentation means parameters are essentially the characteristics of the terminal used, especially the list of devices operating (speaker and touch screen) within their limits, the availability of each of them and a whole set of constraints on the transmission of information through them: dimension constraints such as screen size, processing time constraints and constraints on the constitution of information depending on the chosen modality. For the presentation conditions parameters, we suggest applying the three gesture functions identified by (Cadoz, 1994) to multimedia presentation to describe the gesture interaction possibilities:

– epistemic constraints, linked to getting to know the environment: taking into account the level of ambient noise, which is recorded by the microphone and the level of ambient light, even the vibrations in the case of an HMD on an airplane, at least if it has the correct sensors;

– ergotic constraints linked to the environment transformation: sound level and light level beyond which the system cannot go in order to not disturb the environment, which can be easily pictured in an office environment or a field of operations involving several other systems and users;

– semiotic constraints, linked to the emission of information for the environment, with the speech rate quantity and quality, which, for example, can turn out to be too intense for the communication environment.

Along this line, a third set of parameters covers the human aspects falling within the scope of adapting to the user. It is first and foremost a question of adapting to the user's physical abilities, in case of a disability, that is of constraints on the communication channel operations, but also in general, with the constraints and preferences with regard to the level of use of these channels: if the audio channel is already in use, for example. It is then a question of adapting to the user's roles and individual preferences. It is finally a question of human factors as universal preferences, with physiological factors, linguistic factors and cognitive factors. The first are linked to the nature of the modalities: in sound generation, such as a loud warning beep, the system has to know that the higher pitchers are more strident than the lower ones, and that the louder the warning the better the chances that it is perceived (but the more stressful it becomes). For visual generation, the system can be led, as we saw in the interpretation (paragraph 6.1.2), to implement criteria from the Gestalt theory perceptive grouping or even to take into account color theory observations, to use the color red, that is the color that is most quickly perceived by a human being, for urgent messages. Whatever the modality, the system can use notions of *salience* and *pregnance*: a *salient* element is an element that stands out from the others because of the unique properties, for example the only blue element in a visual scene is easily found; a *pregnant element*, that is which has been previously repeated to the point that the user's memory has recorded it, is also more easily perceived.

The linguistic factors fall within the scope of the user's lexical, prosodic, syntactic, semantic and pragmatic preferences. The system can adapt to these preferences by aligning itself on the user's uses, that is using the same terms, the same sentence structures, with a similar use of language. At the level of natural dialogue in natural language, it is also a question of applying the Gricean maxims when determining the message, minimizing the risks of ambiguity and anticipating the ambiguity generation conditions (e.g. the system can avoid a pronoun anaphora when there are several possible antecedents), it can go so far as to avoid indirect and composite speech acts, at least in simple dialogues where the task is prioritized above all. It is also a question of using the information structure, especially to highlight part of the information and apply simple rules of cohesion and coherence, for example a relevant use of connectors, taking into account the previous messages.

As for the cognitive factors, it is the set of elements that were the focus of section 2.1: taking into account human short-term memory limits, mental representation abilities or even user attention management. It is a question, for example, of not putting the system in a position in which it tries to attract the user's attention in several directions at once. In general, a principle can consist of using what has already correctly worked.

If the system observes that a message has had a positive and efficient influence when generated visually rather than orally, it can choose to use it again with the same type of generation in similar conditions. Finally, the parameters mentioned are managed as follows:

– parameters arising from the application domain, the task and the user model: the levels of urgency and criticality, the self-descriptive information (structure and quantitative information) and the multimedia presentation preference constraints specific to the type of task or the task itself;

– parameters calculated by the dialogue manager: pragmatic values and forces, labels such as those linked to emotions, cohesion and coherence indications, emphases, constraints and preferences with regard to linguistic terms and dialogue management;

– parameters decided by the multimedia presenter based on constraints at other levels: ordering the information to be presented, for example depending on the levels of urgency, the way to dissociate information into several presentation phases, the way to dissociate information over several communication channels, the levels of emphasis of each piece of information, for example depending on its criticality and the ways in which to emphasize it.

9.2. Multimedia presentation pragmatics

9.2.1. *Illocutionary forces and values*

An important aspect is found within the use of the illocutionary values and forces imposed by the dialogue manager. In interpretation and in generation, as well as the message's semantic content, we also have, as we have seen an illocutionary value that refers to the dialogue act carried out by the utterance (or the presentation), a value that can be "saying that", "telling to" or "asking", or a combination of several of these acts (composite act) that is linked to an underlying intention. We can give this value the importance of a force, that is the level of intensity to which the value must be transmitted. The way it is presented, for example a warning, depends on the illocutionary value: if the sole goal is to give information, due to a "saying that", the information can be presented in a neutral manner, which will be very different from the manner chosen for a "telling to", which is the same as giving the user an order to take the warning's significance into consideration, according to an operational mode matching an "inciting to act". The dialogue system can require a confirmation of the message's reception. Thus, we can distinguish an act that consists of information without any specific indication on the rest of the interaction, an act that consists of informing by asking for acknowledgment, which dialogue boxes sometimes do by including the "OK" and "cancel" buttons, which the user has to click on if he wants the task to progress. For these two examples, the notion of act is useful, and thus shows that automatic generation can carry out similar strategies to automatic interpretation. The alert example can thus take on the form of a composite act with a "saying that"

clarifying the nature of the issue and a "telling to" ordering the user to react. As the acknowledgement example, it combines a "saying that" covering the information presented with an "asking" covering the acknowledgment: "OK" is a positive answer to this question while "cancel" is a negative answer.

9.2.2. *Perlocutionary forces and values*

These examples that incite or force the user to react in a certain way are close to the notion of perlocutionary act that aims to have certain effects on the user's mental states (paragraph 7.1.1). What illocutionary values cannot explicitly do through concrete acts, perlocutionary values can help the system, for example the ECA, to manifest a behavior that helps to generate a certain effect on the user. Managing a perlocutionary value is complex, especially when it is accompanied by a certain force. In any case, it is up to the multimedia presenter to find a form of expression for the message, which renders the perlocutionary aim correctly. It can be a specific prosody, or an expression or attitude on the ECA's part that expresses expectation, indicating that it is expecting a reaction from the user. In our example, the system can answer "I do not have any seats left on the train for Paris" to the user's request, which is a simple "saying that" with no other perlocutionary goal than to inform. However, the system can also answer "an hour before it leaves, I don't have any seats left for the Paris train", which has the same illocutionary value with a slight overtone that warns the user that he should have done this earlier and a perlocutionary value aiming to modify the user's beliefs on how to reserve a ticket. In a similar vein, "I do not have any seats for Paris left, only for Massy-Palaiseau" carries a perlocutionary value, which tries to modify the user's initial goal by inciting him to consider an alternative goal, but not expressing it openly, which would have been "are you ready to change your destination to Massy-Palaiseau?".

In natural language, managing the perlocutionary value is close to inference management and planning in dialogue. In other modalities, such as in an HMI, it can be to simply make clear the different possibilities that the user has following one of his actions. In an HMI, buttons are pressed, text is typed in areas that resemble input fields and table cells are clicked on: we know that any displayed element has a function, and we explore this function with the means offered by the keyboard-mouse interaction. Thus, the multimedia presenter managing the HMI at the same time as the HMD has to take into account the HMI element functions in the different presentation phases: for each element involved in the previous or current action, it has to know the input interaction possibilities, eventually block them (grayed out button, table cell displayed in a certain color) and, if needed, let the input manager module know.

This strategy consists of anticipating the user's future actions depending on a specific perlocutionary value and can have numerous complex consequences. When the HMD system asks the user a question, and expects an answer from a set of clearly

identified alternatives, when it incites the user to talk about a new topic and when it presents information in a specific order, and thus incites the user to refer to this information, it leads to a reduction in the interaction possibilities. The user's next utterance can thus almost be predicted. Thus, the speech recognition and linguistic analysis modules benefit greatly from being warned. Taking into account the difficulties in open domain speech recognition, several speech models can be involved: a generalist model can be used at the beginning of the dialogue, but a model oriented toward numbers, dates and values can be used when the user has to answer a question asked by the system and concerning such data. In the same way, following an information presentation in a significant order, the user can be led to use mentional references such as "the first", "the second" and "the last two" or even quantified expressions such as "each…" or "all the…". The multimedia presenter must thus be aware that it is highlighting and ordering, especially when this order was not clarified by the dialogue manager (this is the case when information is simply displayed in a left to right manner), in the same way as it has to be aware when it aggregates information and can thus lead to group references rather than to individual elements. The point is to warn not only the speech recognition module to adapt the speech model, but also the semantic analysis module and the reference resolution module, to let them know of the logic underlying the aggregation's order. When the system is about to reach the point of understanding the user's utterance, it then has all the elements to identify the right referent.

9.3. Processes

9.3.1. *Allocation of the information over communication channels*

We can see with the spread of human factors and the complexity of pragmatic aspects at the generation level that the presentation of multimedia information, if it is implemented according to a deep model, involves numerous parameters and complex, often interdependent processes. Among them, a key process for multimodal dialogue is the allocation of information to be transmitted on the communication channels. The first step in this process consists of taking into account the constraints, the second is taking into account the preferences and the third, which can potentially lead to new decisions, consists of linking the components of the information thus allocated, that is to clarify the links between the modalities.

The first step is prioritizing the constraints. It is a question of taking into account the constraints that are inherent to the information (visual modality for a geographical map), the constraints linked to the terminal (if it does not have any speaker, all the utterances in natural language will have to be displayed in writing, with the morphological and spelling constraints this involves), the constraints linked to the presentation environment (a noisy atmosphere can lead the system to forego a vocal modality) as well as the constraints linked to the user's abilities and roles.

The next step consists of taking into account a set of rules on:

– the urgency or criticality of the message: if one of the parameters is at a high level, it might be required to use all the communication channels;

– the message content: choice of the channel best able to render the information or simultaneous use of several channels if the information is rather complex;

– the communication act: a single channel for a simple act, two channels for a composite act;

– interaction history: prioritize the use of a channel that has already been used;

– user preferences: use a single channel if the user has expressed a preference in that way;

– human factors: allocate in a long display time a very large piece of information for which knowledge acquisition requires maintained attention; take into account the user attention if the system can detect it (in case of sustained attention, the system does not need to make any specific effort to explicitly allocate the information over communication channels).

The third stage, which consists of clarifying a link between the modalities, comes back to the notion of deixis (see paragraph 5.1.4), with terms such as the demonstrative adverb "here". When part of the information is displayed and the other part is verbalized, it can be that the user does not draw a link between them spontaneously. The danger then is that he might consider that the system is providing him with two distinct messages: the two parts of the information can be of different natures and not depend on each other, contrary, for example, to a video in which the soundtrack and the image are immediately seen as two pieces of the same information due to their temporal synchronization. It can thus be helpful for the system to add an indication to the semantic content to be transmitted, which helps the user make a link. This indication is carried by a modality and reminds the user of the existence of the other piece of information. It can be a visual icon indicating that the system is speaking. It can be a deictic referring expression that relies on visual context or an utterance such as "on the geographical map you can see the trains going to Paris" or "the train changing at Meudon is flashing". Generating such utterances requires several processing stages, such as:

– the dialogue manager generates the presentation query "make the Meudon-Paris itinerary clear to the user";

– the multimedia presenter chooses a realization that is visual (display of the journey on a geographical map, with a flashing aspect so that the itinerary is easy to distinguish from the other itineraries displayed), and audible ("saying that" speech act to provide the user with the identity of the itinerary displayed), with the generation of a deixis so that the user can bring both realizations together;

– the multimedia presenter asks the natural language generator to materialize the multimodal deixis by indicating the nature of the display that has been chosen;

– the natural language generator chooses to carry out a multimodal reference, “the flashing one” and builds a simple sentence around this reference, “the Meudon-Paris itinerary is the flashing one”;

– the multimedia presenter sends this utterance in a synchronous manner to the text-to-speech module and carries out the flashing.

Another possibility would be to choose the demonstrative sentence “here is the Meudon-Paris itinerary”. This sentence is shorter and more efficient, but the link between “here” and the flashing is less clear than in “the flashing one”.

9.3.2. *Redundancy management and multimodal fission*

If the multimedia presenter wishes to accentuate the presentation of an object such as the Meudon-Paris itinerary, it can decide to display a text message “Meudon-Paris” on the given itinerary at the same time as it generates the oral utterance “here is the Meudon-Paris itinerary”. In that case, the choice falls within the scope of *redundancy*. The point is that if one of the communication channels is not working correctly, the other can help compensate. Moreover, as an increasing amount of information is generated, there are more chances for the user to perceive it. In the same way, as the information is increasingly marked, there are more chances for the user to remember it, as we saw with the notion of pregnance. However, redundancy is not necessarily a systematic advantage: too many messages do not incite the user to maintain his attention on the issue in question. As it is often said, too much information kills the information. Moreover, too many messages increase the processing and thus reaction time, and if they are not well managed, can lead the user not to draw a link between the different materializations of the same information. In conclusion, redundancy should only be used in situation in which it is obvious that they are redundant. Moreover, we should abstain from using redundancy within a single communication channel, for example by adding the generation of an oral utterance to that of a sound, because both of them occupy the audio channel that ends up overloaded.

Redundancy management and information allocation over communication channels falls within the scope of a single issue, that of multimodal fission. At the level of audio or visual signals, it focuses on directing the information to the appropriate communication channel depending on its nature. Presenting a video involves transmitting the soundtrack in the audio channel and the image on the visual channel, which is a multimodal fission directed by constraints. On the level of semantics, it is to dissociate the information content over several modalities so as to better manage its complexity by obtaining simplified monomodal messages. One example falling within the scope of multimodal fission directed by preferences consists of displaying the part of the information that requires maintained attention and verbalizing the part that only requires selective attention. At the level of pragmatics, it is dissociation of the act of dialogue

of the message over several modalities to obtain simpler dialogue acts, as in the case when we dissociate the components of a composite act to turn it into simple acts. In this case again, it is multimodal fission, that allows us to progress to the three levels identified for multimodal fusion that have equivalents in generation.

9.3.3. *Generation of referring expressions*

The example of "the flashing one" involves several reference phenomena mentioned in Chapter 6: the mention of a property, here given as a relative, which singles out the object and allows the system to identify it, as well as the use of a pronoun that is deictic and anaphoric, deictic because it refers in a deictic manner to an object that has been focalized and anaphoric because it takes up from an antecedent its nominal head, that is "itinerary". All these phenomena give us an idea of the processes (Mellish *et al.*, 2006) that a referring expression generator must manage. A referring expression must enable the user to identify the referent in a non-ambiguous manner. To this end, a now classic algorithm, the incremental algorithm (Reiter and Dale, 2000), finds the properties that allow it to single out the targeted object from the other objects, prioritizing certain properties for certain types of object, following in these near universal preferences. This algorithm has been extended and adapted many times to take into account any type of property including spatial relations between several objects, to manage a consistency criterion between the properties used to refer in a grouped manner to several references or even to take into account contextual markers, especially when the properties chosen are vague ("long journey") and gradable (Kopp *et al.*, 2008; Krahmer and Van Deemter, 2012).

The linguistic challenges for future research studies are a better management of salience, redundancy, sets of referents, referents other than objects, a better use of vague properties, relation managements between more than two objects and a better control over the involuntary generation of ambiguities. In a dialogic framework, it is also a challenge to integrate into the process the ability of joint construction of a reference with the hearer (Krahmer and Van Deemter, 2012). There are more psychological challenges that can be found, as for automatic generation in general, in collaboration with psycholinguists, through, for example, experiments aiming to characterize the notion of salience, and in taking into account human factors, that is in applying to reference the principles that were developed above. From a technical point of view, the current algorithms combine knowledge representation because of adapted logic, research into graphs, constraint satisfaction and context modeling. They are increasingly complex, and as is often the case in NLP and HMD, currently explore combinations of symbolic approaches and approaches based on corpus data.

9.3.4. *Valorizing part of the information and text-to-speech synthesis*

At the end of the output processing chain is the valorizing of part of the information (whether it is visual, in natural language or in any other way) and the materialization processes, especially ECA and text-to-speech synthesis. To highlight the part of the information, the process uses the same principle as that of the allocation of information over communication channels: it starts by taking the constraints into account, then by managing a set of rules modeling the preferences. The constraints are the same as previously stated; for example, the sound level beyond which the system cannot go in spite of the desire to highlight the part of the message, which can be translated by a prosodic accentuation. The rules rely on the message's content, the dialogue act, with, for example, a specific intensity for a "telling to", human factors and individual preferences expressed by the user. The highlighting in itself can take on a variety of shapes, especially for a message in natural language. It can involve information structure, syntactic constructions such as cleft sentences and topicalizations, and a variety of prosodic processes (a pause before and after the emphasized element, for example). At the level of ECA management, specific renditions are implemented. When it comes to data such as a geographical map or a table of numbers, any use of colors, element relative size and Gestalt theory criteria are to be considered.

In a natural dialogue in natural language, the rendition of the system's voice is an essential aspect, which can discourage the user as we saw in paragraph 1.3.2 with the *uncanny valley* issue. In general, and in a rather schematic way, we actually get a system to pronounce any kind of text or utterance with a quality close to that of a human who does not understand the semantic content. We still lack a better ability to take contextual aspects into account, for example the rendition of nuances, which reveal a fine understanding. C. d'Alessandro in (Chaudiron, 2004) shows that the evolution of the text-to-speech system has gone down the following path: type 1 systems, able to regurgitate prerecorded messages; type 2 systems, managing simple messages built with a set vocabulary through concatenation methods; type 3 systems, able to carry out a genuine synthesis from a text; and type 4 systems with a visual component. The paradox of text-to-speech used in HMD is that often this process is seen as a module to be run at the end of the chain, i.e. little concerned with the prosodic and linguistic analysis processes; when the current realizations show that to pronounce an utterance correctly, the system must understand its meaning and master prosody so that it can, if needed, align itself with the user. The phenomena of accentuation, prosodic prominence, intonation and rhythm are part of the preoccupation in text-to-speech, see Chapter 11 of (Cohen *et al.*, 2004). We have emphasized how much the phase of *how to say it* was important in a dialogue, for reasons of coherence and cohesion. Moreover, once the word sequence making up the message in natural language has been decided upon, it is mostly prosody that will allow us to control its synthesis. (Theune, 2002) shows, for example, that a deep specification of prosodic directives is essential and suggests a generation model that is not necessarily directed at dialogue,

in which several processes happen in a chain to enrich the text to be spoken into an annotated text, which will then serve for the text-to-speech phase.

The alignment takes on an important part in this context, especially after works on prosodic as well as lexical alignment (Brennan and Clark, 1996) and in general along all the dimensions of language (Pickering and Garrod, 2004). Recently, (Branigan *et al.*, 2010) emphasized not only how much the alignment was involved in HMD, but also to what point it is different from alignment in human dialogue. Alignment with a machine is seen as a strategy for communication to work. Thus, alignment is seen as one of the many major challenges for HMD.

9.4. Conclusion

In addition to vocal returns, the system can manifest reactions and answers through visual returns or even gestures if it can materialize itself with an avatar displayed on screen. The task can also involve the presentation of information to the user. The generation of all these messages at the system's output involves many choices on its part: what information should be highlighted? What part of the information should be rendered visually? What part should be rendered orally? This chapter has discussed the necessary parameters for such choices and on the automatic generation techniques of multimodal and linguistic messages. We also find here the linguistic preoccupations mentioned in input processing as well as the cognitive aspect preoccupations such as taking into account human factors to generate adapted messages.

Chapter 10

Multimodal dialogue system assessment

Whether it happens at the very end of the design, on the final system or during the design itself, on prototypes or system modules, assessment is meant to measure performances, compare these performances with those of existing systems and find the strong and weak points. If the means allow it, the latter can lead to taking up a design phase again to improve the system. While it is often disparaged, maybe because it touches on raw nerves, assessment can also bring a precious point of view and usable methods to design and implementation. When talking of the assessment of referring expression automatic generation algorithms, (Krahmer and Van Deemter, 2012) observe that the first works were not clear about the parameters used in the algorithms, and that it is when the assessments were carried out that the researchers had to lay out all their cards, and describe their favorite parameters. Assessment campaigns also helped bring together researchers focusing on the same issues and thus participate in the general dynamics. However, it comes with many constraints that can discourage some. For example, comparing several systems requires the projection of results for each system toward a common formalism that will allow for comparisons. Yet, this projection can be very costly in terms of time and does not bring anything to the system itself.

In this chapter, we will not present the assessment methods specific to each module in a dialogue system, such as the assessment methods of a speech recognition engine, those of an anaphora resolution module or even the perceived quality of an ECA, see Chapter 9 of (Garbay and Kayser, 2011), or of text-to-speech synthesis, without forgetting the assessment of models and model derivation processes used in design-time architectures (see section 4.2). Each domain, even the very vast domain of interpretation or that of dialogue management, has its own criteria, which refer to the techniques used and which are well beyond the scope of this book. We present the characteristics

of an approach aiming to assess an HMD system, with the oral interaction and multimodal interaction specificities, taking into account unbiased aspects (speaking turns, average duration of utterances, number of refusals) and biased aspects (interview of a user on his feelings, filling out questionnaires), which are also the topic of descriptive and inferential statistical analyses, see section 2.2 and (Jurafsky and Martin, 2009, p. 872).

We first present the methods that are currently used, in oral and multimodal HMD as well as in HMI (section 10.1), which lead us to underline the weak points of the assessment and present the challenges for the years to come (section 10.2) as well as a few paths that could be followed, especially for multimodal dialogue (section 10.3).

10.1. Dialogue system assessment feasibility

We can see an increasing number of articles being published on oral and multimodal dialogue system assessment. Assessment paradigms are suggested, and they are broader and broader and more and more complex, notably covering metrics, user tests or questionnaire analysis methods, with questionnaires filled out by subjects after they have used the system. These efforts are relevant and should be applauded, but we should not forget some recurrent observations that are particularly true.

First observation: contrary to information extraction, speech recognition or syntactic analysis systems, HMD systems often remain at the level of research prototypes which are hard to make and operate correctly, and which are very sensitive to user behavior. Apart from a few marginal, recreational examples, there is not a single system that has been sold to the public and used in a profitable manner by a large amount of people. In other words, scalability is still a major issue in HMD and the assessments carried out limit themselves to research prototypes or professional systems that are so task specific (such as military systems) that they only concern an extremely small number of users. HMD assessment is thus reduced to a very limited area, which does mitigate its scope somewhat without questioning its use.

Second observation: soon there might be as many assessment methodologies as systems themselves. This is not an issue per se, but it does lead to questions. We can, in particular, wonder about the validity of an assessment method proposed by the designers of a system, which is only meant to assess this single system, and the method itself is assessed by its application to the system in question. This description might appear excessive, but it does reflect a certain reality, or at least get close to it. This situation cannot be avoided due to the small number of systems and, faced with each system's progress, to the need to take into account aspects that exiting assessment methodologies do not deal with. Thus, we end up proposing or extending an assessment methodology to be able to assess the progress of a new system. The speed

of the technological progress only increases this phenomenon. HMD assessment thus seems to always be a step behind its goal.

Third observation: the assessment is used not only to improve a particular system's development (using measurements, diagnostics and satisfaction questionnaires) but also to compare one system with another. Several campaigns have been started, and what has emerged is that it is very hard to compare several dialogue systems, even if they have been designed for comparable application contexts, for example train time or hotel information, which are two widely used applications. In this aspect, the HMD domain seems to give rise to a more delicate issue than the other NLP domains and contributes to the fragile image that is attached to its assessment.

Faced with these observations, we can wonder how feasible it is to assess HMD. With this goal in mind, this section goes over the main issues and a few methodologies that we believe to be promising. The question of feasibility strikes us as a key issue which has not been discussed enough and for which we will try and provide a few thoughts. The criticism we have just carried out with the previous observations does not prevent us, later on, from suggesting ways to better take multimodality into account in the existing paradigms. Section 10.2 thus presents a few multimodality extension possibilities for methodologies meant for the oral, and for the few methodologies already focusing on multimodality. The example "put that there" that we used in paragraph 6.2.2 will help us illustrate them.

10.1.1. *A few assessment experiments*

Let us take up, as we did in the scenarios of section 3.1, a generic designer: whether it is the last design stage or the middle stage, our system designer will inevitably wonder about the issue of assessment. Given the simplifications that were carried out on the linguistic and pragmatic theories, is the system performing sufficiently? In general, whether there were any simplifications or not, did the system have satisfactory reactions? Is each module fulfilling its role? Is the architecture relevant for the processes carried out? Does the system's behavior match the original idea that was the basis for the directions given to the Wizard of Oz?

Obviously, the first idea that comes to mind once the system is operational is to carry out user tests. It is often the point when the designer's morale is put to the test: between the subjects that do not understand how to use the microphone and the related push-to-talk button or pedal; those that do not manage to use the touch screen; those that do not control their action on the material and even go so far as to destroy it; those that repeat each gesture three times or repeat each bit of sentence three times for fear that the system might have missed a part of it; those that express themselves so spontaneously that their sentences are teeming with incisions, relative subordinates, hesitations or corrections; those that are so intimidated by the system

that they express themselves in a telegraphic style; and mostly those that go beyond the predefined applicative framework. One word (which had not been imagined at the beginning) is enough for the despairing "I did not understand..." statement to be generated, a situation that was experienced when assessing the OZONE project, with the multimodal train reservation system which integrates a complete modeling of trains, train stations, timetables, but whose lexicon did not contain the "tomorrow" that one of the assessors uttered.

Faced with this observation, the designer then carries out a training course on human-machine interaction with its subjects, on what the spontaneous specific dialogue is, on the way in which the system operates and especially on the applicative domain and its scope. In the end, a subject directly generated valid utterances and the system operated much better. If this is not the case, a training session (which does not count for the assessment) can be considered. But then, what about the assessment? By repeating sentence examples that the system can process, the subjects are obviously led to utter these same sentences, and it is hard to assess the *spontaneous* aspect of the dialogue. It is the same when the speech recognition module requires learning: training this module on reference sentences repeated by the speaker leads him to get a precise idea of the system's abilities and direct its behavior as a consequence.

However, our designer can now compare the subjects' utterances with the utterances imagined at the beginning. It is sometimes a second test for the designer's morale: by trying to make the system work or simply sticking to the examples produced, the subjects sometimes limit their utterances to very simple sentences, which are devoid of any of the phenomena that were planned for at the beginning and meant to be the point and innovating aspect of the system. As an example, one of the subjects that was recorded in the MAGNET'OZ corpus, see Chapter 13 in (Van Deemter and Kibble, 2002), only generated almost two sentences in half an hour, that is "put that here" and "put that there", which indeed falls within the scope of the expected multimodality, but proves itself to be extremely restrained! Our designer, who wishes to assess his system by testing it, for example, on interesting multimodal reference situations, is thus frustrated to the point that he sometimes incites the subject to generate an utterance close to the utterance that he would assess (a situation that was experienced during the MAGNET'OZ recording). At this point, the experiment conditions are not a worry anymore, and we start taking up the list of initial phenomena to make them go one by one through the system... If this method allows the designer to carry out a diagnostic on his system, it is clear that it does not allow him to generate a satisfactory assessment.

Our designer then attempts another method: obtain the users' impressions as well as all kinds of information on their experience with the system. Whether it is a recorded interview or the filling out, in a free or directed manner, of a questionnaire, the information gathered is often below the designer's expectations: it is too general ("yes, the system answers correctly, but not always", "it's a nice project", "it was

fun"); it lacks precision ("it was a bit slow"); there is a discrepancy compared with the expectations ("I used gestures"); it is irrelevant (a long appreciation of the icon appearance and objects displayed, a criticism of the applicative task or the performance of the expert system integrated in the system). In the end, the designer is faced with anecdotes he will not be able to tell anything from that can be properly used for assessment or to carry out a diagnostic of his system.

For systems that are aimed at specialists, i.e. users who know the task or users who already use a similar system but in versions built on classic graphical user interfaces, sometimes, the worse is yet to come: these specialists have their habits and try and find them again or test them on the new multimodal system. The smallest error, uncertainty, discrepancy between what they define to be normal and what they are faced with is intolerable. They are all the more susceptible to question the point of voice command or multimodal command. In other words, these are the most demanding users you can imagine, and getting assessment elements from them is a risk. This is the field, however, in which the main challenge of HMD is a domain of applied research: with the aim to bring multimodal communication spontaneity to society, we have to go beyond the research prototypes and aim to build systems for the general public or transform existing graphical systems into multimodal systems. In other words, we need to aim for scalability, with all the qualitative and quantitative challenges that it involves (see paragraph 3.2.6).

Taking such aspects into account when building multimodal dialogue systems raises many challenges, and a perfectly trustworthy build is not in the immediate future. Yet, a true assessment is only relevant in genuine operating conditions. This might be the reason why the HMD domain is so different from most of the other NLP domains: the scalability of systems that rely on the written language and implement fewer sensible software components is not as delicate as a dialogue system. Some systems, for example seeking and indexing engines, are even designed, very early on, to process a huge amount of data in genuine use conditions. Their scalability is thus made easier, and assessment can be carried out with stability.

10.1.2. *Human-machine interface methodologies*

Scalability is also mastered in the HMI domain, which can inspire the HMD since the interaction between the human being and the machine is at the heart of the interface. As (Kolski, 1993) and (Grislin and Kolski, 1996) show, an assessment consists of *checking* and *validating* the system. The system is checked to see if it matches the specifications arising from the definition of needs and validated if it matches the needs and respects the application domain constraints. The ergonomic assessment of an HMI consists of ensuring that the user is able to carry out his task due to the system that is offered to him. The notion of usefulness determines if the HMI allows the user to achieve his work goals. This usability accounts for the human-machine interaction

quality and whether it is easy to learn and use. These notions can very well be applied to HMD as well as the ergonomic criteria identified for HMI: social acceptability (socially unacceptable systems would, for example, be systems asking the users nosy questions) and practical acceptability (production, cost and trust constraints). Acceptability is also the field in which we find usability, with criteria such as easy to learn, to memorize, few errors, ability to guide the user, work load management, etc.

Beyond these criteria, the field of HMI offers a set of methods that can be combined to lead to a relevant assessment: representative user opinion analysis, representative user activity analysis (video recordings, observation, use of an eye tracker, physiological measurements), expert judgment, assessment grids covering the lists of qualities for a good system. If these methods can only be used in the case of a running system, others can be considered even at the stage of the system design: expert judgments, theoretical modeling of the interaction (analytical approaches: predictive formal models, quality models and software models). Finally, a third set of methods is dedicated to prior assessment, that is as early as the specification phases, for example, by taking into account the importance of human factors when designing a system or by following the principles of cognitive engineering, especially that which aims to replace the user at the heart of each specification and design stage.

A lot of these methods find no equivalent in HMD. This is the case, for example, of analytical approaches that aim to formally model the user's behavior: if certain behaviors, when faced with an HMI, can be predictable and can be formalized, the issue is more complex in HMD. For that, we only need to look at all the previous chapters. Some methods, however, are easily usable and materialized even more precisely in HMD. For example, when it comes to the classic method of expert judgment, (Gibbon *et al.*, 2000) show, for example, that in HMD, at least three experts are required to identify almost half of the usability problems. The higher the number of experts involved, the higher the number of problems identified, but the more the assessment is costly in time and human resources. More recently, (Kühnel, 2012) presents a set of methods, including not only an expert group test, but also the method of cognitive walkthrough, which consists first and foremost of an analysis of the task by decomposing it into actions: at least one expert follows a path determined to be optimal to solve the task, and checks at each stage that the following stage is accessible for any novice user.

10.1.3. *Oral dialogue methodologies*

More specifically on oral HMD, a certain number of methods have been suggested (Antoine and Caelen, 1999; Devillers *et al.*, 2004; Dybkjær *et al.*, 2004; Walker, 2005; Möller *et al.*, 2007; Kühnel, 2012). These make up a sort of reference framework that contains recommendations to implement user interaction test methods to automatically analyze or semi-automatically analyze the obtained interaction traces, markers to

determine the assessment metrics or even the principles to create and analyze the questionnaires filled out by the users. We thus find a few of the methods used by HMI. Each system assessor can thus pick among this stock to determine the method(s) he will apply. Indeed, a single test seems to be insufficient and a genuine assessment seems to need to bring several tests together. The evaluation campaigns (EVALDA/MEDIA: assessment methodology for understanding within and outside of the dialogue context), the work groups (MADCOW group, speech understanding group, GdR I3) and the various European project consortia widely use this principle. When several systems are involved and the assessment is comparative, the operational rules can be defined so as to better control the assessment quality. The challenge assessment campaign with its management crossed with the designer roles of the systems involved (Antoine, 2003) is an example.

The methodology's main propositions are each accompanied with an original idea that is meant to simplify the implementation of a type of test by providing it with a means to be operationalized in a specific context. The paradigm of the MADCOW group (Hirschman, 1992) thus provides us with the notion of template that characterized the minimum and maximum answers to a query and thus make its assessment more rigorous. The PARADISE paradigm, Paradigm for Dialogue System Evaluation (Walker *et al.*, 2001), focuses on the maximization of the user's satisfaction and suggests to try and satisfy the task as a reference. Another original idea example (López-Cózar Delgado *et al.*, 2003) suggests assessing a system by automatically generating test user utterances, that is modeling the user's behavior, including his mistakes. In France, this method was taken up in the SIMDIAL paradigm (Allemandou *et al.*, 2007), in which the deterministic simulation of a user allows us to automatically assess the system's dialogic abilities, notably thanks to the notion of disturbing phenomenon, which, like the noise in the Wizard of Oz of (Rieser and Lemon, 2011), allows us to introduce protestations or rephrasing requests that will allow us to assess the system's general behavior and robustness. Moreover, the Data–Question–Response (DQR) methodology, see notably the chapter by J. Zeiliger *et al*. in (Mariani *et al.*, 2000), introduces the principle of questioning the system about the point to be assessed, with the advantage of thus displacing the focus of the assessment from the data to the question, and thus not focusing on the answers or reactions of the system (*black box* method, which does not require exploring the system's internal structures, but lacks precision), nor on the system's semantic structures (*transparent box* method, which is precise and easily leads to a diagnostic but requires to have reference semantic representations). The system still needs to be able to answer the questions Q of DQR. The adapted paradigm Demand–Control–Response-Result-Reference (DCR, Antoine and Caelen, 1999), minimizes this issue by replacing the question with a control that is a simplification or a rephrasing of the initial user query. The PEACE paradigm (automatic understanding assessment paradigm: "Paradigme d'Evaluation Automatique de ComprEhension"), see the chapter by L. Devillers *et al*. in (Gardent and Pierrel, 2002),

which contributes to the original idea of modeling the dialogue history with a paraphrase, allows us to stay within the black box model while allowing for an assessment of context understanding.

10.1.4. *Multimodal dialogue methodologies*

In the multimodal HMD context, propositions are far from being as relevant. The PROMISE paradigm, Procedure for Multimodal Interactive System Evaluation (Beringer *et al.*, 2002), is presented as an extension of PARADISE to multimodality, with principles to affect scores to multimodal inputs and outputs. The proposition in fact remains at a very approximate level, well below the variety of multimodal phenomena. The interesting aspects of the article concern the oral dialogue, with considerations on the level of task completion and the level of user cooperation. The works of N.O. Bernsen and L. Dybkjær, which are still references in the field of multimodal dialogue, are rather disappointing when it comes to assessment. (Bernsen and Dybkjær, 2004) present a methodology meant for a system, with a focus on the method of the questionnaire subsequently filled out by users. The reason given is indeed that the other methods are not yet well established. Unfortunately, the questions in the questionnaire remain within a very superficial level when it comes to multimodality: "did you use the mouse or point onto the screen?", "how was it to do the gestures?", "would you like to be able to do more with gesture? If yes, what?". The answers provided by the users also strike us as very poor in content, especially because one of the authors' conclusions is that the user preferred talking rather than using the multimodal possibilities. As for (Dybkjær *et al.*, 2004), it is more a review of methodologies and projects than a proposition of new methodologies for multimodality: the content of the book remains within general recommendation level. In another register, (Vuurpijl *et al.*, 2004) present a tool called μ*-eval*, for the transcription of multimodal data and a system's assessment. Moreover, the assessment does not concern only the dialogue turns and does not deal with multimodal phenomena. Finally, (Walker *et al.*, 2004) focus on the user models and the (oral) dialogue strategies but barely on the multimodal aspects.

In general, to assess multimodal systems, we cannot reuse the principles applied in the oral system assessment campaigns. Each methodology proposal is only carried out within the framework of a single system, which itself is the system on which the methodology is tested (see our second observation). Instead of focusing on this work, we will take up a few identified issues and methodologies proposed within oral dialogue to check their relevance in multimodal dialogue.

10.2. Multimodal system assessment challenges

10.2.1. *Global assessment or segmented assessment?*

The first question that arises when implementing an assessment procedure is the choice between a global assessment and a segmented assessment (or modular assessment). The *global* assessment considers the dialogue system as a black box and is interested in utterances exchanges, i.e. interaction traces. The assessment then focuses on the relevance of the system's reaction versus the user utterance, on the progress of the task as the exchanges progress, but does not take the architecture or the system's internal functionalities into account. Its implementation falls within the scope of user tests and corpus tests (including the test follow-ups). The assessment itself can consist of a simple subjective analysis of the interaction traces or of the application of unbiased metrics that allows us to obtain figures and results (Walker *et al.*, 2001). In any case, the application of this method to multimodal system creates the issue of the coverage of the tests carried out, i.e., the coverage of the assessed dialogue situations and the coverage of the test corpus. To validate a system based on spontaneous interaction, we will thus have to test a great number of situations to account for the interaction's variability. This issue can already be found in oral dialogue, but the multiplication of parameters in multimodal communication situations gives it even more weight in this case. As an example, object references can take on a variety of linguistic forms, including the various types of pronouns and noun phrases. In multimodal HMD, each of these forms exists and can be matched with a pointing gesture. The variety of pointing gestures thus has to be taken into account, as well as that of visual contexts in which the gestures were generated (see Chapter 6). The combination is thus greater and the coverage of reference situations should follow this scale increase.

Segmented assessment considers the system to be a transparent box and is interested in the internal functionalities and representations. The assessment then focuses on the inputs and outputs of each module. Within an oral dialogue, we can often limit ourselves to the semantic module output and compare the semantic representations obtained with reference representations. The application of this method to multimodal systems generates a few issues, some were already present in oral systems but are exacerbated and others are specific to the introduction of multimodality. Thus, if we focus on the assessment of multimodal understanding, i.e. on the fusion of information recorded at input (knowing that the assessment of the multimodal generation will ask similar questions):

– we can proceed as for the oral and focus on the multimodal semantic representations obtained at the output of the module managing the global semantic analysis, i.e. the module responsible for multimodal fusion. Depending on the system (e.g. if the multimodality covers natural language and conversational gestures or, on the contrary, brings together emotion detection on the user's face with lip reading and natural language analysis), these multimodal semantic representations can vary a lot and

cover a wide variety of phenomena. The main issue is then to determine the reference multimodal representations that several systems will have in common. Specifying exhaustive representations is almost impossible, especially since the technology evolves and makes any specification quickly obsolete;

– we can, on the contrary, consider a multimodal system to be a process of fusion as well as a set of monomodal systems, each characterized by a type of semantic representation. The assessment then concerns, on the one hand, the fusion process and, on the other hand, each monomodal system, with reference representations each time. We will thus have to first specify the reference semantic representations for gesture trajectories, emotion interpretation, etc. The point of this method is to better target the diagnostic, since we can identify which monomodal processing chain is lacking. The inconveniences, obviously, are the multiplicity of indispensable reference representations, as well as the need for a specific assessment method for multimodal fusion;

– on the other hand, if we keep on this track, we can consider that the assessment has to be applied in a modular fashion, that is by using reference representations to assess the outputs of each of the system's modules. The main issue with this approach, beyond a strong dependence on the architecture, is the multiplication of modules and thus that of the matching assessments and representations: lexical, syntactic and semantic representations of natural language, facial expressions, gesture trajectories, etc. A modular assessment is thus an important work load. Another issue: if we multiply local assessments, it becomes difficult to confront measurements to achieve a global system assessment. The diagnostic is more precise, but it comes at the expense of a simple measurement that can help understand the global operation quality. In addition, and this is a key point, the errors of a module can be made up by the performances of another module without having any consequence on the global operation. It is not about compensating the bad running of a module through an exemplar running of another module, but to compensate unavoidable errors with relevant making up procedures. The typical example concerns the speech recognition errors: it is an illusion to hope for a recognition module that would be able to have perfect performances (100% of words perfectly recognized) with a very broad vocabulary, or even a few hundred words. However, if the semantic and pragmatic modules are able to transform semantic and contextual information as recognition constraints, the system will be able to find the pronounced utterance in a sure way, even if the recognition engine's performance is mediocre. In other words, the recognition module assessment has no point, and only the recognition module and the semantic and pragmatic modules interaction assessment counts. The modular assessment is thus not quite as simple to implement nor is it as relevant.

10.2.2. *Should a multimodal corpus be managed?*

A second issue that arises when implementing an assessment procedure is the use of a multimodal corpus. In this case, it is a test corpus, which is obviously different from the corpus previously used to gather phenomena or for a potential machine learning purpose. The system is input with all the situations found in the corpus, and we test either the system's output or its internal semantic representations. If the procedure appears to be clear, it does, however, present a certain number of issues linked to multimodality coding.

At the most *raw* level, a multimodal corpus is a recording of signals captured by the system: an audio signal captured by the microphone, a gesture signal tracked by the touch screen, a video signal tracked by one or more cameras, etc. If such a recording can be ideal to simulate system input, it does not allow for any data manipulation (here we can think of the derivation of examples from an initial example, see paragraph 3.2.2) and is hard to characterize in terms of phenomena. To do this, the corpus annotation, i.e. the passage from a raw level to an interpreted level, is often an indispensable operation. Yet, the annotation of a multimodal corpus creates issues due to the nature of the recorded signals. Contrary to speech that can relatively simply be transcribed in a relatively simple and unbiased manner into written sentences, gesture and other modalities cannot be transcribed simply, which we have seen in paragraph 6.1.3. It remains, however, that these technical issues should not let us lose sight of the fact that we cannot do without corpus use in HMD.

10.2.3. *Can we compare several multimodal systems?*

A third issue concerns the implementation of a *comparative assessment* procedure. The point is to compare several dialogue systems with similar abilities on the same type of application. But in the existing works that limit themselves to oral dialogue, the comparison focuses rarely on comparing an oral dialogue system with another type of reference system, for example a written dialogue system. It would, however, be interesting to assess the contribution of speech as a source of communication improvement between the user and his machine, or a source of improvement in the task management efficiency. The question arises especially when it comes to a multimodal dialogue. Often the multimodal ability is presented as an asset compared with the linguistic ability: multimodality is presented as being more efficient, quicker, more precise and direct, especially for referring actions that allow a direct access to the objects (without going through complex and potentially ambiguous spatial descriptions). A comparative assessment procedure should thus include oral systems as well as multimodal systems. Moreover, and this is especially true in professional fields, multimodality is also presented as an asset compared to the classical graphical user interface based on windows, menus, icons and buttons. Accessing the objects

displayed on the screen is indeed the basis of both of them. A comparative assessment procedure should, therefore, include graphical user interfaces and multimodal systems. The efficiency, speed and precision aspects become the many measurements that allow us to compare a multimodal system with a graphical user interface for the same task (not all of the multimodal systems, however, are made after a first graphical version).

A questionnaire was produced for the people undergoing a user test, an assessment to diagnose, make a global assessment, segmented assessment, assessment based on the system's reaction or comparing its internal representations with reference representations. Most of the methods suggested for oral dialogue appear to be applicable to multimodal dialogue, even if a certain amount of care should be taken when implementing them. Among the approaches that strike us as very hard to apply to the multimodality are the comparative assessment and the passage on a corpus. Even if the latter is useful to test or train, it remains hard to imagine how to reuse a *ready-made* multimodal corpus with annotations and which could be adapted to the assessment of a system for which it has not been designed. We can, however, hope for future development in multimodal corpora, especially if the synergy we can observe emerging between the corpus-assessment campaign pair, such as the synergy we can observe in EVALDA/MEDIA (Devillers *et al.*, 2004), reaches multimodality.

Finally, the impressions that come from this are that the HMD domain and the multimodal dialogue, do not lend themselves well to the other domains of NLP assessment. Methodology that is precise, trustworthy, objective, complete, independent of system, modalities and tasks is still an unattainable Grail. The acronyms of the methodologies proposed echo this impression: PARADISE, PEACE, PROMISE, etc. Luckily, the issues that have arisen and the questions asked, and the aspects processed, even if they do not lead to a unanimous assessment methodology, greatly contribute to the improvement of implementation processes and of dialogue system test processes.

10.3. Methodological elements

Among the methodological aspects we have mentioned, we consider a few here, which are as many elements for the methodology of the assessment of multimodal dialogue systems. These propositions concern the user's level of expertise, the questionnaires for the user test subjects, the implementation of procedures for the multimodal systems (this is about studying the feasibility of multimodal DQR–DCR), with the implementation of paraphrases of a multimodal dialogue history (we would like to study the feasibility of a multimodal PEACE). We leave the other ideas suggested within the framework of the oral dialogue aside, such as user behavior simulation, for example, with automatic generation of test utterances. The multimodal behavior of a user still does not remain well-known and such a simulation can still appear delicate for now. This is a challenge for multimodal dialogue.

10.3.1. *User expertise and system complexity*

We will focus here on the *level of expertise* of the user with regard to information provided by the user and useful for processing the questionnaire as well as for a better analysis of his behavior with respect to the system. This preoccupation cannot appear compatible with the natural dialogue in natural language: from the moment when the targeted system has to allow spontaneous dialogue, any user who knows how to talk is an expert. In fact, HMD systems only get close to natural dialogue, and especially when they integrate multimodal tracking devices, they remain the first and foremost computational systems for which the more or less expert user opinions are essential. In the methodologies mentioned in paragraphs 10.1.3 and 10.1.4, this level of expertise is often limited to the opposition between novice and expert, see chapter by L. Devillers *et al*. in (Gardent and Pierrel, 2002). Following the Danish Dialogue Project, two indicators are sometimes mentioned: first, for the level of expertise related to the task domain and second for the level of expertise related to the specific system, but both indicators remain binary, either novice or expert (Dybkjær *et al.*, 2004). The perspectives given at the end of these articles underline the need for a more refined scale to better analyze the data provided by the tests.

Classic AI works have long suggested typologies at various levels, each level characterized by a type of behavior when facing a problem. In (Dreyfus and Dreyfus, 1986), there are five levels of expertise detailed:

– novice, who limits himself to applying set and deterministic rules;

– advanced beginner, notices situations and acts according to past experiences;

– competent, able to carry out strategic plans;

– proficient, able to intuitively eliminate plans which have no prospect of succeeding;

– expert, who intuitively moves in a space of reasoning.

Providing the users with these definitions, thus enabling them to determine their own level of expertise can be slightly complicated, and has its own risks. One solution is to ask them questions directly, such as “did you recognize a usual communication situation?” and then deduce, bit by bit, their level of expertise.

In the case of professional users such as soldiers who know their task perfectly and have a more or less in-depth knowledge of the system, the level of expertise is a crucial piece of information. We could imagine an operator with a status of expert for a voice-controlled system but a novice status for the multimodal system devoted to the same task. The analogy with the number of hours logged by a pilot is immediate, since it matches the same issue: instead of, or in additional to, a label such as *competent* or *expert*, it is interesting to use the number of hours that the user has spent with the system. For a plane as for a dialogue system, this number of hours implicitly

includes a training and learning period, and a single measurement can be enough. If the time spent in training is considered to be a relevant measurement, for example if the user tests have two sets of subjects: one which has accelerated training and the other in-depth training, we can either keep two indicators (two numbers of hours) or carry out a calculation with different weighting for the training and the subsequent use. According to the same principle, we can draw a distinction between the time spent with the real system and with the Wizard of Oz. The analogy then relies on the number of hours that a pilot spends in a simulator and the level of expertise includes a new form of weighting. And again, according to this principle, we can dissociate the time potentially spent training the speech recognition engine, time which allowed the user to get a precise idea of the operational sentences. Or consider that repeating prepared utterances is a type of training, and thus include it in the training period.

In any case, it is possible to go beyond the novice–expert dichotomy, as long as we know how a more precise piece of information can be used. Indeed, using several levels of expertise allows us to compare several users with similar levels (equal in case of a discrete measure, in the same interval in case of a continuous measure as with previous calculations), as well as several users of varied levels. The dialogue system assessment in that case includes new indicators, such as:

– Do users of the same level have the same multimodal behavior and do they have the same problems using the system?

– Is a user of a higher level more efficient, quicker and more precise in his use of multimodality? Does he cover a wider set of phenomena? Does he exploit more functionalities of the system?

Finally, if we come to managing a level of expertise not for the whole multimodal system but only in the multimodal aspects of the interaction, another problem arises: that of the legitimacy of an independent measure of the system which would thus be valid for any multimodal system. This level of multimodal expertise could be based on the use of one or more multimodal systems and would require taking into account an indicator of the complexity of each system. Taking up the previous analogy, 100 h of flight experience on a small tourist plane is not the same as 100 h on a fighter plane, even if both types of planes have similar points when it comes to the way they are piloted. The weight of the time with a system complexity index is a possible solution. This complexity index should depend on the following aspects: the number of words of vocabulary in the system, the number of recognized syntactic constructions, the number of processed gesture types, the number of speech acts, the number of devices (terminals and screens), the number of languages identified, the number of possible exchanges and potentially the number of primitives to the application task. We find the set of multimodal system functionalities, and this makes the implementation of an absolute index hard. A solution thus consists of choosing a reference system and calculating, as a ratio, the gap between this system and the system considered.

10.3.2. *Questionnaires for users*

The questionnaire is the main source of subjective information. It helps broaden the scope of the assessment and carry out a diagnostic, not only of the system but also of the user training content. We will consider here the questionnaires offered for oral dialogue and complete them by adding multimodal preoccupations.

Some general questions can be taken up from the questionnaires by N.O. Bernsen and L. Dybkjær. They are essentially questions on the user's impressions, the system's perceived use and the possible ways to improve it: "was it easy or difficult to use the system? why?"; "could you understand what he said?"; "could he follow what you wanted to talk to him about?"; "what do you think of his behaviour on the screen?"; "what was bad about your interaction with the system?"; "what was good about your interaction with the system?"; etc. (Bernsen and Dybkjær, 2004). We can quickly add more questions to these questions, which focus on modalities and multimodality:

– Can you understand the gestures generated? Did the system understand your gestures? What did you think of the force-feedback's operation?

– Were the generated messages correctly created? In a synchronous manner? With no inconsistency or gap? Did the system correctly link your gestures to what you said? Did it understand your entire messages?

The general questions of the PARADISE paradigm can also quickly be extended to multimodality with the same principle, which consists of answering with a satisfaction index between 1 and 5:

– the system was easy to understand (oral and multimodal aspects);

– the system understood what I said (we can replace *say* by *generate*);

– I got the information I asked for (always valid);

– the pace of interaction with the system was appropriate (always valid, but very important in multimodal settings since it includes potential problems of temporal synchronization between the different modalities – in other words, the answer is not interpreted in the same way depending on whether the system is oral or multimodal);

– I knew what I could say at each point (we can replace *say* by *generate*);

– the system clearly explained what it had understood (always valid);

– the system's questions or suggestions helped me (always valid).

We can also suggest some examples of questions more or less focused on multimodal phenomena spread over four categories:

1) Questions focusing on interaction conditions: did you feel free when you were talking? Did you generate gestures? When did you use the force feedback device? Did the system correctly manage the devices at its disposal? Did you feel constrained in their use? And in that case, which of the devices constrained you?

2) Questions on input processing: did the system understand your way of communicating with it properly? Did it appear to be sensitive to your emotions? In general, did the system take your expressiveness into account?

3) Questions on output management: what did you think of the way in which the answers were presented (orally, graphically or both)? Did the avatar seem natural? When the avatar generated a gesture, did it seem relevant?

4) Questions on the relationships between input and output managements: did you feel there was a coherence between your messages and the messages generated by the system? Did the expressiveness appear better in input or output?

As we can see, it is not hard to specify a set of questions covering multimodality management. They seem to us at least to be more precise than those by (Bernsen and Dybkjær, 2004) and should avoid overly vague answers. They cannot make us forget that nothing can replace the spontaneous writing of a text describing the subject's impressions, especially before any question can be asked. Indeed, a question directs the subject's attention to the mentioned topic and can thus introduce a bias in relation to his initial impressions. Even the order of the questions can have an influence on the answers. The specification of the questionnaire should thus be done under the direction of a psychologist, which was not the case for PARADISE, PEACE or this one, which is one of the challenges of this methodological aspect.

10.3.3. *Extending DQR and DCR to multimodal dialogue*

J. Zeiliger *et al*. in (Mariani *et al.*, 2000) have elected a *black box* type of methodology that allows them to carry out a system diagnostic, a methodology that relies on generic tests to assess the understanding of an isolated utterance. The contextual aspects have been neglected (we will return to that with PEACE), but it was the price to pay for achieving a simple and well-defined methodology. The principle is to carry out occasional assessments, each of them focused on a specific phenomenon. Thus, in the DQR materialization, the occasional assessment becomes a question Q which the system asks and it allows us to check that it has understood the original request D properly. One of the examples given concerns anaphora resolution, with a request, question and subsequent answer:

– D = "take the first street on the right and follow it for 300 yards" (initial utterance as it was given to the system to help the task progress);

– Q = "follow the street on the right?" (question addressed to the system after the utterance D and meant to assess the proper understanding of D);

– R = "yes" (system's answer showing that the anaphora was correctly understood and making the assessment positive).

The authors specified seven levels characterizing the scope of the questions asked. Let us take these levels up here by indicating each time how to extend the paradigm to be able to use it in a multimodal dialogue.

– Level 1 = *explicit information*. It is the marking of an explicit piece of information in the utterance, the point is to test the good understanding of the literal utterance given the great variability of spontaneous language. The examples given by the authors limit themselves to taking up part of the utterance and asking the user to confirm this part has been understood: D = "you will take at right after the white buildings with blue shutters" then Q = "white shutters?" or "blue shutters?". The extension of this principle to multimodality consists of asking questions on elements on the multimodal utterance. With D = "put that there" + gesture in (x_1, y_1) + gesture in (x_2, y_2), we can test the tracking abilities of the multimodality by asking the following questions Q: "that?" + gesture in (x_1, y_1); "put there?" + gesture in (x_2, y_2); "put that?" + gesture in (x_2, y_2); "put that there?" + gesture in (x_2, y_2) + gesture in (x_1, y_1), etc. The process can appear naive but it allows the system to simply test the proper matching of gestures with referring expression, which is an important part of the multimodal fusion. Specific attention is given to temporal synchronization between the words pronounced and the gestures generated. Thus, a temporal delay between "that" and the gesture in question Q could lead, depending on the system, to a positive answer, which would reflect its robustness in multimodal matching even when the production conditions deviate, or, on the contrary, to a negative answer reflecting the system's inability to go beyond a certain temporal gap.

– Level 2 = *implicit information*. This level concerns the resolution of anaphora, ellipses, gaps and other implicit information that can be salvaged at a syntactic and semantic level. An example would involve: D = "give me a ticket for Paris and one for Lyon too" and Q = "ticket for Lyon?". The reference resolution is one of the main aspects of spontaneous multimodality, so a multimodal DQR will have to obviously account for it. Thus, if we take up D to be the multimodality universal primitive, "put that there" with two pointing gestures, the questions Q could introduce precisions on the referents, starting, for example, from the mention of their category and going so far as to give them a unique identifier as it is managed by the system: "put this object?" + gesture in (x_1, y_1); "put this file?" + gesture in (x_1, y_1); "put 'submis.tex'?" (no gesture); "put obj_{4353}?" (no gesture), etc. The assessment procedure thus includes natural language paraphrase of a multimodal reference. What remains simple for the deictic gesture is much less simple for other types of co-verbal gestures. Let us imagine, for example, that "put that there", or rather "move that there" so the example is not to complicated, is accompanied with a single gesture going from the object to be moved and ending at the destination. According to a first hypothesis that takes up the presentation of paragraph 6.2.2, this gesture trajectory is considered to be the materialization of the necessary transition between pointing at an object and pointing at a location. In this case, only the ends of the curve are used during the semantic analyses: the point (x_1, y_1) then the object present in this point or its immediate vicinity are unified with "that", and the point (x_2, y_2) is unified with "there". In other words, we

return to the previous case. According to a second hypothesis, the trajectory is considered to be a combination of these two pointing gestures with an illustrating co-verbal gesture providing a characteristic on the movement action, that is the path (or points of passage) to follow. The trajectory is analyzed from a temporal point of view (curve generated in a regular manner, with no significant stop) and a structural point of view (arc), before being unified with "move", that is interpreted as a movement path. If we wish to test this multimodal system's functionality, we only need to ask an additional question Q: "follow this trajectory?" or "move along these points of passage?", by taking up the full gestures in both cases. The only inconvenience applies to all the DQR methodologies, that is the need for the system to process such questions.

– Level 3 = *inference*. This is about the construction of the utterance's full meaning, the difficulty lies in the identification of allusions, an identification that calls upon common sense reasoning and pragmatic inferences. With D = "I would like a return ticket for Paris", the authors suggest Q = "would like ticket?". This aspect is independent from the communication modalities and is still valid for the multimodal dialogue.

– Level 4 = *illocutionary act interpretation*. Here we enter the levels of dialogue, with a first aspect concerning speech acts and the system's ability to identify the correct type of act, even in the case of an indirect act. With D = "a ticket for Paris", which can follow a question or match an initial request, the question Q = "is this a request?" allows the system to assess the act it has identified. In multimodal settings, we find dialogue acts, especially gesture acts, which we talked about in the previous chapters, and we can consider the following questions Q: "is this gesture a request?" + gesture; "does this gesture accompany speech?" + gesture; "is the multimodal utterance a request?" (to test multimodal fusion at a pragmatic level), etc. With this aspect and the aspects detailed in levels 1 and 2, we have gone over the main issues, which can appear when processing multimodal dialogue input.

– Level 5 = *intention recognition*. This is meant to determine the underlying intentions or goals the user's utterances have and is thus at a deeper level than level 4. The principle is to explicitly question the intentional states with questions Q such as: "does the user know, does the user want... ?". Such intentional states are independent of the communication modalities, and the DQR extension to multimodality does not lead to any change at this level.

– Level 6 = *relevance of the answer*. The point of the assessing question is rather broad here, because it is meant to test the relevance of the system's answers. The aspects covered should thus be the linguistic (and thus multimodal) abilities found in the answers, how they match the user's initial utterance, the application knowledge, the communication means, the user's profile, etc. In the chapter by J. Zeiliger *et al*. in (Mariani *et al.*, 2000), the examples of questions Q are the following: "is this question aggressive?"; "is this question necessary?"; "is this proposition possible at this moment?". These examples question both the form and the content of the answer. In multimodal dialogue, we thus would need to add all the aspects linked to output modality, i.e. to the choices the system made when determining the content and form

of the multimodal answer. Thus, possible examples for Q are: “is the choice of the output modality(ies) relevant?”; “is the message overloaded?”; “is the message redundant?”; “is the message synchronized?”; “is the information presented relevant?”. These questions look at all the main issues that can arise in output in a multimodal dialogue. They, however, integrate metalinguistic aspects that are not generally implemented in the conceptual model and systems’ lexicon. More than an extension principle of DQR to multimodality, the feasibility of this level 6 seems unrealistic.

– Level 7 = *strategy relevance*. This level tests the dialogue strategy’s quality, i.e. if it was efficiently driven and if it has succeeded. In fact, the questions cover not only the dialogue strategy but also the task management strategy: “is the client happy?”; “are there too many indirect confirmation questions?”; “is the user’s level of upset due to the strategy?”. Slowness, the number of incidences and the reasons of a rupture can also be questioned. These aspects are independent of the modalities and so there is nothing to add and we finally have a full-fledged multimodal DQR.

As we have mentioned in paragraph 10.1.3, the other materialization of this methodology is the DCR paradigm (Antoine and Caelen, 1999), which replaces the assessing question by a control C, thus minimizing the issue of the system’s ability to answer a question that is at times metalinguistic. The control consists of a simplification or a reformulation of the initial user request. Thus, taking up some of the previous examples, we can have multimodal controls intervene in the following: “put ‘submis.tex’ in (x_1, y_1)”; “move obj_{4353} by (x_1, y_1) to (x_2, y_2)”; “move obj_{4353} according to the points of passage (x_3, y_3), (x_4, y_4), (x_5, y_5)”. Going from a multimodal DQR to a multimodal DCR requires the paraphrase to be done in a simple and non-ambiguous manner for the multimodal references, with the description of the spatial coordinates in natural language. The other aspects do not present any specific problem, or at least no more a problem than the passage from DQR to DCR.

10.3.4. *Towards other assessment methods*

The principles of PEACE, presented in the article by L. Devillers *et al*. in (Gardent and Pierrel, 2002), are the reformulation of the dialogue history into a single sentence, the use of this sentence for a contextual assessment of the current utterance and the use of reference semantic representations. We have already mentioned the difficulty in applying this principle to multimodal dialogue, and thus the idea of reformulating the history is what we are focusing on here.

Dialogue history modeling is a recurring issue in HMD and is particularly complex in multimodal dialogue (Landragin, 2004). As we mentioned in Chapter 6, the dialogue history must keep both the referent identifiers (to use them when interpreting an anaphora) and the mentions used to refer to it (to interpret mentioning references or metalinguistic expressions as well as to interpret ellipses, and especially noun ellipses). In a multimodal system, it is the same thing and multimodal referring forms

have to be kept, as well as the state of the visual scene at each stage, thus leading to at least a linguistic history, a gesture history and a visual history. A reference calling on the modalities used and the user's memories can then be interpreted, for example "the object which I have just pointed at", "the two objects grouped together a bit further away", "the left one", "the one which used to be on the right", "the last one". These referring expressions themselves show that the paraphrase of a multimodal dialogue history is an impossible task, or it would be achieved only by simplifications such that the bias introduced would take any plausibility away from the assessment. Indeed, the only paraphrase process that can be automatized is the systematic use of referent identifiers, and this solution seems to be more destructive in multimodal dialogue than it is in oral dialogue: it pushes away all the multimodal aspects. It thus appears hard to apply the principles of PEACE to a multimodal dialogue.

The assessment of multimodal dialogue systems turns out to be more complex than that of oral systems (which was already quite difficult), especially when multimodality is considered to be the complementary association of natural language and other communication modalities on which language relies. In this chapter, we have suggested a few methodological building blocks, especially an extension of the DCR and DQR paradigms to multimodality. Various aspects still have to be studied to achieve a methodology covering the field naturally occupied by multimodal dialogue. An aspect concerns one of the paths currently explored to simplify multimodal system design, that of model-driven engineering (see section 4.2). In addition, when there will be enough multimodal dialogue systems, we believe it is useful to return to the challenging assessment method. Its principle, whether it is the stage of utterance derivation from a set of initial utterances or the exchange of roles between different designers, does strike us as relevant to multimodal dialogue.

10.4. Conclusion

The assessment of human-machine dialogue is not characterized by the efficiency, objectivity or consensus that can be observed in other natural language processing fields. The systems are designed for a specific task, which makes any standardized or comparative assessment difficult. Moreover, technological progress makes many assessment paradigms obsolete and thus causes them to be multiplied. This chapter synthesizes the existing methods and suggests a set of reflections around multimodality assessment in systems with a strong linguistic component.

Conclusion

As (Luzzati, 1995, p. 6) already wrote almost 20 years ago, HMD is still "a new type of communication which has to be invented almost at the same time as the material that supports it, and whose nature depends on the abilities that the machine is given, both in understanding mechanisms as well interaction of generation mechanisms". We have seen that in spite of the technical progress and, for example, the increasing use of machine learning algorithms, the amount of work required to create an HMD system depends on the abilities that are considered for this system, whether they are the abilities to process various signals, NLP abilities, the logical reasoning abilities or the abilities to create and render visual messages. We have emphasized in this book the importance of a multidisciplinary method that integrates experimentation, corpus studies and theory confrontation in their HMD application. One aspect of this multidisciplinary method, assessment, is so complex that research efforts have still to be made. We have emphasized the point of dynamic management of speech turns, the importance of prosody and semantics (rather than syntax) in the linguistic analysis process, as well as the key roles of the reference resolution, dialogue act identification and planning processes. Through an example that at first appeared to be simple, a train timetable information task, we have provided a panel of current techniques and challenges for closed-domain dialogues.

Without analyzing the challenges mentioned in each chapter, let us discuss the four sets of challenges detailed in section 1.3. The first set of challenges brings together the theoretical challenges, with the exploration of linguistic theories and their adaptation to HMD, an adaptation that can happen by questioning certain historical anchors such as the breakdown into syntax, semantics and pragmatics. We have emphasized the importance of works interfacing between two disciplines, with the now self-evident example of the interface between linguistic theories and computational implementations. This intermediary position has the advantage of helping to identify which linguistic works can have relevant applications and contribute to HMD with theoretical abilities and an overall view that is sometimes desirable. But it is also very uncomfortable: the researcher at the border between two disciplines does not

contribute to linguistic theories (only to their applications) and does not create computational development (only prerequisites). Thus, there is no direct result that can be promoted, and his or her contribution, to a formal model or to paths for an implementation, can be easily criticized: only an implementation comes with *proof* and can resist criticism. With this book, we hope to have helped to show how these border works remain indispensable.

The second set of challenges focuses on the panel of abilities expected in a system. We have shown that widespread understanding abilities were the key to relevant exchanges and a realistic dialogue. Moreover, as (Cole, 1998, p. 200) underlines it, for closed-domain system, it is first and foremost robustness and real-time aspects that have to be improved, as well as the system's abilities to lead the user (without railroading him) into an operational path.

The third set of challenges covers the methodological and technical challenges around system design. We have given examples of complex work flows, involving the implementation not only of run-time architectures, but also, something that is not as common, of design-time architectures that are indispensable in allowing a certain flexibility in the development as well as an amount of reusability. Moreover, we have shown that the amount of work required to design an HMD system goes beyond that of a doctoral dissertation and, thus, a team is now necessary. The elements of this team are different depending on their professions, just like what is being done in other computer science fields.

Finally, the last set of challenges is that of facilitating computational development with toolkits, and maybe some day middleware and hardware cards for NLP, automatic understanding and dialogue management. Today, creating an HMD system that can keep up the state of the art in most of its functionalities and add an innovative aspect is a genuine challenge, unless we have a work environment that provides a continuously updated platform. The generalization of this type of platform and the means of easiness we have mentioned will be a major advance for the field of HMD. It will enable us not only to significantly increase the research result speed, but also to carry out more trustworthy assessments that will be more comparable than they are now.

References

ABBOTT B., *Reference*, Oxford University Press, Oxford, 2010.

ABEILLÉ A., *Les grammaires d'unification*, Hermès-Lavoisier, Paris, 2007.

ALLEMANDOU J., CHARNAY L., DEVILLERS L., LAUVERGNE M., MARIANI J., « Un paradigme pour évaluer automatiquement des systèmes de dialogue homme-machine en simulant un utilisateur de façon déterministe », *Traitement Automatique des Langues*, 48(1), pp. 115–139, 2007.

ALLEN J.F., PERRAULT C.R., « Analysing Intention Utterances », *Artificial Intelligence*, 15, pp. 143–178, 1980.

ALLEN J.F., SCHUBERT L.K., FERGUSON G., HEEMAN P., HWANG C.H., KATO T., LIGHT M., MARTIN N., MILLER B., POESIO M., TRAUM D.R., « The TRAINS Project: A Case Study in Defining a Conversational Planning Agent », *Journal of Experimental and Theoretical Artificial Intelligence*, 7(1), pp. 7–48, 1995.

ALLWOOD J., TRAUM D.R., JOKINEN K., « Cooperation, Dialogue and Ethics », *International Journal of Human-Computer Studies*, 53, pp. 871–914, 2000.

ANTOINE J.Y., Pour une ingénierie des langues plus linguistique, mémoire d'Habilitation à Diriger des Recherches, University of South Brittany, Vannes, 2003.

ANTOINE J.Y., CAELEN J., « Pour une évaluation objective, prédictive et générique de la compréhension en CHM orale : le paradigme DCR (Demande, Contrôle, Résultat) », *Langues*, 2(2), pp. 130–139, 1999.

ASHER N., GILLIES A., « Common Ground, Corrections, and Coordination », *Argumentation*, 17, pp. 481–512, 2003.

ASHER N., LASCARIDES A., « Indirect Speech Acts », *Synthese*, 128(1–2), pp. 183–228, 2001.

ASHER N., LASCARIDES A., *Logics of Conversation*, Cambridge University Press, Cambridge, 2003.

AUSTIN J., *How to do things with words*, Oxford University Press, Oxford, 1962.

BAKER M.J., Recherches sur l'élaboration de connaissances dans le dialogue, mémoire d'Habilitation à Diriger des Recherches, University of Lorraine, 2004.

BEAVER D.I., CLARK B.Z., *Sense and Sensitivity: How Focus Determines Meaning*, Blackwell, Oxford, 2008.

BELLALEM N., ROMARY L., « Structural Analysis of Co-Verbal Deictic Gesture in Multimodal Dialogue Systems », *Progress in Gestural Interaction. Proceedings of Gesture Workshop*, York, United Kingdom, pp. 141–153, 1996.

BERINGER N., KARTAL U., LOUKA K., SCHIEL F., TÜRK U., « PROMISE – A Procedure for Multimodal Interactive System Evaluation », *Proceedings of the LREC Workshop on Multimodal Resources and Multimodal Systems Evaluation*, Las Palmas, pp. 77–80, 2002.

BERNSEN N.O., DYBKJÆR H., DYBKJÆR L., *Designing Interactive Speech Systems. From First Ideas to User Testing*, Springer Verlag, Berlin, 1998.

BERNSEN N.O., DYBKJÆR L., « Evaluation of Spoken Multimodal Conversation », *Proceedings of the Sixth International Conference on Multimodal Interfaces*, Penn State University, USA, pp. 38–45, 2004.

BEUN R.J., CREMERS A.H.M., « Object Reference in a Shared Domain of Conversation », *Pragmatics and Cognition*, 6(1/2), pp. 121–152, 1998.

BILANGE E., *Dialogue personne-machine : modélisation et réalisation informatique*, Hermès, Paris, 1992.

BLANCHE-BENVENISTE C., *Approches de la langue parlée en français* (seconde édition), Ophrys, Paris, 2010.

BOLT R.A., « Put-That-There: Voice and Gesture at the Graphics Interface », *Computer Graphics*, 14(3), pp. 262–270, 1980.

BOBROW D.G., KAPLAN R.M., KAY M., NORMAN D.A., THOMPSON H., WINOGRAD T., « GUS, A Frame-Driven Dialog System », *Artificial Intelligence*, 8, pp. 155–173, 1977.

BRANIGAN H.P., PICKERING M.J., PEARSON J., MCLEAN J.F., « Linguistic Alignment between People and Computers », *Journal of Pragmatics*, 42, pp. 2355–2368, 2010.

BRENNAN S.E., CLARK H.H., « Conceptual Pacts and Lexical Choice in Conversation », *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6), pp. 1482–1493, 1996.

BROERSEN J., DASTANI M., VAN DER TORRE L., « Beliefs, Obligations, Intentions, and Desires as Components in an Agent Architecture », *International Journal of Intelligent Systems*, 20(9), pp. 893–919, 2005.

BUNT H., « Multifunctionality in dialogue », *Computer Speech and Language*, 25, pp. 222–245, 2011.

CADOZ C., « Le geste canal de communication homme-machine. La communication instrumentale », *Techniques et Sciences Informatiques*, 13(1), pp. 31–61, 1994.

CAELEN J., XUEREB A., *Interaction et pragmatique. Jeux de dialogue et de langage*, Hermès-Lavoisier, Paris, 2007.

CARBERRY S., *Plan Recognition in Natural Language*, The MIT Press, Cambridge, 1990.

CHAROLLES M., *La référence et les expressions référentielles en français*, Ophrys, Paris, 2002.

CHAUDIRON S. (ed.), *Evaluation des systèmes de traitement de l'information*, Hermès-Lavoisier, Paris, 2004.

CLARK E.V., *First Language Acquisition* (seconde édition), Cambridge University Press, Cambridge, 2009.

CLARK H.H., *Using Language*, Cambridge University Press, Cambridge, 1996.

CLARK H.H., SCHAEFER E.F., « Contributing to Discourse », *Cognitive Science*, 13, pp. 259–294, 1989.

CLARK H.H., WILKES-GIBBS D., « Referring as a Collaborative Process », *Cognition*, 22, pp. 1–39, 1986.

COHEN M.H., GIANGOLA J.P., BALOGH J., *Voice User Interface Design*, Addison-Wesley, Boston, 2004.

COHEN P.R., LEVESQUE H.J., « Intention is Choice with Commitment », *Artificial Intelligence*, 42, pp. 213–261, 1990.

COHEN P.R., PERRAULT C.R., « Elements of a Plan-Based Theory of Speech Acts », *Cognitive Science*, 3, pp. 177–212, 1979.

COLBY K.M., WEBER S., HILF F.D., « Artificial Paranoia », *Artificial Intelligence*, 2, pp. 1–25, 1971.

COLE R. (ed.), *Survey of the State of the Art in Human Language Technology*, Cambridge University Press, Cambridge, 1998.

CORBLIN F., *Les formes de reprise dans le discours. Anaphores et chaînes de référence*, Rennes University Press, Rennes, 1995.

CORBLIN F., *Représentation du discours et sémantique formelle*, PUF, Paris, 2002.

DENIS A., Robustesse dans les systèmes de dialogue finalisés. Modélisation et évaluation du processus d'ancrage pour la gestion de l'incompréhension, PhD Thesis, University of Lorraine, 2008.

DENIS A., « Generating Referring Expressions with Reference Domain Theory », *Proceedings of the 6th International Natural Language Generation Conference*, Dublin, Ireland, pp. 27–35, 2011.

DE RUITER J.P., CUMMINS C., « A Model of Intentional Communication: AIRBUS (Asymmetric Intention Recognition with Bayesian Updating of Signals) », *Proceedings of the 16th Workshop on the Semantics and Pragmatics of Dialogue*, Paris, pp. 149–150, 2012.

DESSALLES J.L., *La pertinence et ses origines cognitives*, Hermès-Lavoisier, Paris, 2008.

DEVILLERS L., MAYNARD H., ROSSET S., PAROUBEK P., MCTAIT K., MOSTEFA D., CHOUKRI K., CHAMAY L., BOUSQUET C., VIGOUROUX N., BÉCHET F., ROMARY L., ANTOINE J.Y., VILLANEAU J., VERGNES M., GOULIAN J., « The French MEDIA/EVALDA Project: The Evaluation of the Understanding Capability of Spoken Language Dialog Systems », *Proceedings of the Fourth International Conference on Language Resources and Evaluation*, Lisbon, Portugal, pp. 2131–2134, 2004.

DREYFUS H.L., DREYFUS S.E., *Mind over Machine: The Power of Human Intuition and Expertise in the Area of the Computer*, Basil Blackwell, Oxford, 1986.

DUERMAEL F., Référence aux actions dans des dialogues de commande homme-machine, PhD Thesis, University of Lorraine, 1994.

DYBKJÆR L., BERNSEN N.O., MINKER W., « Evaluation and Usability of Multimodal Spoken Language Dialogue Systems », *Speech Communication*, 43(1–2), pp. 33–54, 2004.

EDLUND J., HELDNER M., GUSTAFSON J., « Utterance Segmentation and Turn-Taking in Spoken Dialogue Systems », dans B. Fisseni, H.C. Schmitz, B. Schröder, P. Wagner (eds), *Computer Studies in Language and Speech*, Peter Lang, pp. 576–587, 2005.

ENJALBERT P. (ed.), *Sémantique et traitement automatique du langage naturel*, Hermès-Lavoisier, Paris, 2005.

FRASER N.M., GILBERT G.N., « Simulating Speech Systems », *Computer Speech and Language*, 5, pp. 81–99, 1991.

FUCHS C., *Les ambiguïtés du français*, Ophrys, Paris, 2000.

FUNAKOSHI K., NAKANO N., TOKUNAGA T., IIDA R., « A Unified Probabilistic Approach to Referring Expressions », *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Seoul, South Korea, pp. 237–246, 2012.

GAONAC'H D. (ed.), *Psychologie cognitive et bases neurophysiologiques du fonctionnement cognitif*, PUF, Paris, 2006.

GARBAY C., KAYSER D. (eds), *Informatique et sciences cognitives. Influences ou confluence ?*, Ophrys, Paris, 2011.

GARDENT C., PIERREL J.M. (eds), Dialogue : aspects linguistiques du traitement automatique du dialogue, *Traitement Automatique des Langues*, 43(2), Hermès-Lavoisier, Paris, pp. 1–192, 2002.

GIBBON D., MERTINS I., MOORE R. (eds), *Handbook of Multimodal and Spoken Dialogue Systems*, Kluwer Academic Publishers, Dordrecht, 2000.

GINZBURG J., *The Interactive Stance*, Oxford University Press, 2012.

GOROSTIZA J.F., SALICHS M.A., « End-User Programming of a Social Robot by Dialog », *Robotics and Autonomous Systems*, 59(12), pp. 1102–1114, 2011.

GRAU B., MAGNINI B. (eds), Réponses à des questions, *Traitement Automatique des Langues*, 46(3), Hermès-Lavoisier, Paris, pp. 1–233, 2005.

GRICE H.P., « Logic and Conversation », dans P. Cole, J. Morgan (eds), *Syntax and Semantics*, Vol. 3, Academic Press, pp. 41–58, 1975.

GRISLIN M., KOLSKI C., « Evaluation des Interfaces Homme-Machine lors du développement des systèmes interactifs », *Technique et Science Informatiques*, 15(3), pp. 265–296, 1996.

GRISVARD O., Modélisation et gestion du dialogue oral homme-machine de commande, PhD Thesis, University of Lorraine, 2000.

GROSZ B.J., SIDNER C.L., « Attention, Intentions and the Structure of Discourse », *Computational Linguistics*, 12(3), pp. 175–204, 1986.

GUIBERT G., *Le « dialogue » homme-machine. Un qui-pro-quo ?*, L'Harmattan, Paris, 2010.

GUYOMARD M., NERZIC P., SIROUX J., « Plans, métaplans et dialogue », *Actes de la quatrième école d'été sur le traitement des langues naturelles*, downloaded on the authors' web page, 1993-2006.

HARDY H., BIERMANN A., BRYCE INOUYE R., MCKENZIE A., STRZALKOWSKI T., URSU C., WEBB N., WU M., « The AMITIÉS System: Data-Driven Techniques for Automated Dialogue », *Speech Communication*, 48, pp. 354–373, 2006.

HARRIS R.A., *Voice Interaction Design: Crafting the New Conversational Speech Systems*, Morgan Kaufmann, San Francisco, 2004.

HIRSCHMAN L., « Multi-Site Data Collection for a Spoken Language Corpus: MADCOW », *Proceedings of the* DARPA *Speech and Natural Language Workshop*, New York, USA, pp. 7–14, 1992.

HORCHANI M., Vers une communication humain-machine naturelle : stratégies de dialogue et de présentation multimodales, PhD Thesis, Joseph Fourier University, Grenoble, 2007.

ISSARNY V., SACCHETTI D., TARTANOGLU F., SAILHAN F., CHIBOUT R., LEVY N., TALAMONA A., « Developing Ambient Intelligence Systems: A Solution based on Web Services », *Automated Software Engineering*, 12(1), pp. 101–137, 2005.

JOKINEN K., MCTEAR M.F., *Spoken Dialogue Systems*, Morgan and Claypool, Princeton, 2010.

JÖNSSON A., DÄHLBACK N., « Talking to a Computer is not like Talking to your Best Friend », *Proceedings of the Scandinavian Conference on Artificial Intelligence*, Tromsø, Norway, 1988.

JURAFSKY D., MARTIN J.H. (eds), *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* (seconde édition), Pearson, Upper Saddle River, NJ, 2009.

KADMON N., *Formal Pragmatics*, Blackwell, Oxford, 2001.

KAMP H., REYLE U., *From Discourse to Logic*, Kluwer, Dordrecht, 1993.

KENDON A., *Gesture: Visible Action as Utterance*, Cambridge University Press, Cambridge, 2004.

KERBRAT-ORECCHIONI C., *L'implicite*, Armand Colin, Paris, 2012.

KNOTT A., VLUGTER P., « Multi-Agent Human-Machine Dialogue: Issues in Dialogue Management and Referring Expression Semantics », *Artificial Intelligence*, 172, pp. 69–102, 2008.

KOLSKI C., *Ingénierie des interfaces homme-machine. Conception et évaluation*, Hermès, Paris, 1993.

KOLSKI C. (ed.), *Interaction homme-machine dans les transports*, Hermès-Lavoisier, Paris, 2010.

KOPP S., BERGMANN K., WACHSMUTH I., « Multimodal Communication from Multimodal Thinking. Towards an Integrated Model of Speech and Gesture Production », *International Journal of Semantic Computing*, 2(1), pp. 115–136, 2008.

KRAHMER E., VAN DEEMTER K., « Computational Generation of Referring Expressions: A Survey », *Computational Linguistics*, 38(1), pp. 173–218, 2012.

KÜHNEL C., *Quantifying Quality Aspects of Multimodal Interactive Systems*, Springer, Berlin, 2012.

LAMEL L., ROSSET S., GAUVAIN J.L., BENNACEF S., GARNIER-RIZET M., PROUTS B., « The LIMSI ARISE System », *Speech Communication*, 31(4), pp. 339–354, 2003.

LANDRAGIN F., *Dialogue homme-machine multimodal. Modélisation cognitive de la référence aux objets*, Hermès-Lavoisier, Paris, 2004.

LANDRAGIN F., « Visual Perception, Language and Gesture: A Model for their Understanding in Multimodal Dialogue Systems », *Signal Processing*, 86(12), Elsevier, Amsterdam, pp. 3578–3595, 2006.

LANGACKER R.W., *Foundations of Cognitive Grammar. Theoretical Prerequisites*, Stanford University Press, Stanford, 1987.

LARD J., LANDRAGIN F., GRISVARD O., FAURE D., « Un cadre de conception pour réunir les modèles d'interaction et l'ingénierie des interfaces », *Ingénierie des Systèmes d'Information*, 12(6), pp. 67–91, 2007.

LEVINSON S.C., *Pragmatics*, Cambridge University Press, Cambridge, 1983.

LÓPEZ-CÓZAR DELGADO R., ARAKI M., *Spoken, Multilingual and Multimodal Dialogue Systems. Development and Assessment*, Wiley and Sons, Chichester, 2005.

LÓPEZ-CÓZAR DELGADO R., DE LA TORRE A., SEGURA J.C., RUBIO A.J., « Assessment of Dialogue Systems by Means of a New Simulation Technique », *Speech Communication*, 40, pp. 387–407, 2003.

LUPERFOY S., « The Representation Of Multimodal User Interface Dialogues Using Discourse Pegs », *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, USA, pp. 22–31, 1992.

LUZZATI D., *Le dialogue verbal homme-machine*, Masson, Paris, 1995.

MARIANI J., MASSON N., NÉEL F., CHIBOUT K. (eds), *Ressources et évaluations en ingénierie de la langue*, AUF et De Boeck University, Paris, 2000.

MARTIN J.C., BUISINE S., PITEL G., BERNSEN N.O., « Fusion of Children's Speech and 2D Gestures when Conversing with 3D Characters », *Signal Processing*, 86 (12), Elsevier, Amsterdam, pp. 3596–3624, 2006.

MCTEAR M.F., *Spoken Dialogue Technology: Toward the Conversational User Interface*, Springer-Verlag, London, 2004.

MELLISH C., SCOTT D., CAHILL L., PAIVA D., EVANS R., REAPE M., « A Reference Architecture for Natural Language Generation Systems », *Natural Language Engineering*, 12, pp. 1–34, 2006.

MITKOV R., *Anaphora Resolution*, Longman, London, 2002.

MOESCHLER J., *Argumentation et conversation. Eléments pour une analyse pragmatique du discours*, Hatier, Paris, 1985.

MOESCHLER J. (ed.), « Théorie des actes de langage et analyse des conversations », *Cahiers de linguistique française*, 13, University of Geneva, 1992.

MÖLLER S., SMEELE P., BOLAND H., KREBBER J., « Evaluating Spoken Dialogue Systems According to De-Facto Standards: A Case Study », *Computer Speech and Language*, 21, pp. 26–53, 2007.

MUSKENS R., « Combining Montague Semantics and Discourse Representation », *Linguistics and Philosophy*, 19(2), pp. 143–186, 1996.

OVIATT S.L., « Ten Myths of Multimodal Interaction », *Communications of the ACM*, 42(11), pp. 74–81, 1999.

PAEK T., PIERACCINI R., « Automating Spoken Dialogue Management Design Using Machine Learning: An Industry Perspective », *Speech Communication*, 50, pp. 716–729, 2008.

PICKERING M.J., GARROD S., « Toward a Mechanistic Psychology of Dialogue », *Behavorial and Brain Sciences*, 27, pp. 169–226, 2004.

PIERREL J.M., *Dialogue oral homme-machine*, Hermès, Paris, 1987.

PINEDA L., GARZA G., « A Model for Multimodal Reference Resolution », *Computational Linguistics*, 26(2), pp. 139–193, 2000.

POESIO M., TRAUM D.R., « Conversational Actions and Discourse Situations », *Computational Intelligence*, 13(3), pp. 309–347, 1997.

PRÉVOT L., Structures sémantiques et pragmatiques pour la modélisation de la cohérence dans des dialogues finalisés, PhD Thesis, Paul Sabatier University, Toulouse, 2004.

REBOUL A., MOESCHLER J., *Pragmatique du discours. De l'interprétation de l'énoncé à l'interprétation du discours*, Armand Colin, Paris, 1998.

REICHMAN R., *Getting Computers to Talk Like You and Me*, The MIT Press, Cambridge, 1985.

REITER E., DALE R., *Building Natural Language Generation Systems*, Cambridge University Press, Cambridge, 2000.

RIESER V., LEMON O., *Reinforcement Learning for Adaptive Dialogue Systems. A Data-driven Methodology for Dialogue Management and Natural Language Generation*, Springer, Heidelberg, 2011.

ROSSET S., Systèmes de dialogue (oral) homme-machine : du domaine limité au domaine ouvert, mémoire d'Habilitation à Diriger des Recherches, University of Paris-Sud, Orsay, 2008.

ROSSET S., TRIBOUT D., LAMEL L., « Multi-level Information and Automatic dialog Act Detection in Human-Human Spoken Dialogs », *Speech Communication*, 50(1), pp. 1–13, 2007.

ROSSI M., *L'intonation, le système du français*, Ophrys, Paris, 1999.

ROSSIGNOL S., PIETQUIN O., IANOTTO M., « Simulation of the Grounding Process in Spoken Dialog Systems with Bayesian Networks », *Proceedings of the 2nd International Workshop on Spoken Dialogue Systems Technology*, Gotemba, Japan, pp. 110–121, 2010.

ROULET E., AUCHLIN A., MOESCHLER J., RUBATTEL C., SCHELLING M., *L'articulation du discours en français contemporain*, Lang, Bern, 1985.

SABAH G., *L'intelligence artificielle et le langage. Tome 2 : processus de compréhension*, Hermès, Paris, 1989.

SABAH G., « The "Sketchboard": A Dynamic Interpretative Memory and its Use for Spoken Language Understanding », *Proceedings of the Fifth European Conference on Speech Communication and Technology*, Rhodes, Greece, 1997.

SABAH G., VIVIER J., VILNAT A., PIERREL J.M., ROMARY L., NICOLLE A., *Machine, langage et dialogue*, L'Harmattan, Paris, 1997.

SACKS H., SCHEGLOFF E.A., JEFFERSON G., « A Simplest Systematics for the Organization of Turn-Taking for Conversation », *Language*, 50(4), pp. 696–735, 1974.

SEARLE J., *Speech Acts*, Cambridge University Press, Cambridge, 1969.

SEARLE J., VANDERVEKEN D., *Foundations of Illocutionary Logic*, Cambridge University Press, Cambridge, 1985.

SENEFF S., « TINA: A Natural Language System for Spoken Language Application », *Computational Linguistics*, 18(1), pp. 62–86, 1995.

SINGH S.P., LITMAN D.J., KEARNS M., WALKER M.A., « Optimizing Dialogue Management with Reinforcement Learning: Experiments with the NJFun System », *Journal of Artificial Intelligence Research*, 16, pp. 105–133, 2002.

SOWA J., *Conceptual Structures. Information Processing in Mind and Machine*, Addison-Wesley, Reading, MA, 1984.

SPERBER D., WILSON D., *Relevance. Communication and Cognition* (seconde édition), Blackwell, Oxford (United Kingdom), Cambridge (USA), 1995.

STOCK O., ZANCANARO M. (eds), *Multimodal Intelligent Information Presentation*, Springer, Heidelberg, 2005.

STONE M., LASCARIDES A., « Coherence and Rationality in Grounding », *Proceedings of the 14th Workshop on the Semantics and Pragmatics of Dialogue*, Poznań, Poland, pp. 51–58, 2010.

TELLIER I., STEEDMAN M. (eds), Apprentissage automatique pour le TAL, *Traitement Automatique des Langues*, 50(3), ATALA, pp. 1–243, 2009.

THEUNE M., « Contrast in Concept-to-Speech Generation », *Computer Speech and Language*, 16, pp. 491–531, 2002.

TRAUM D.R., « 20 Questions on Dialog Act Taxonomies », *Journal of Semantics*, 17(1), pp. 7–30, 2000.

TRAUM D.R., HINKELMAN E.A., « Conversation Acts in Task-Oriented Spoken Dialogue », *Computational Intelligence*, 8(3), pp. 575–599, 1992.

TRAUM D.R., LARSSON S., « The Information State Approach to Dialogue Management », dans J. Van Kuppevelt, R. Smith (eds), *Current and New Directions in Discourse and Dialogue*, Kluwer, Dordrecht, pp. 325–354, 2003.

VAN DEEMTER K., KIBBLE R. (eds), *Information Sharing. Reference and Presupposition in Language Generation and Interpretation*, CSLI Publications, Stanford, CA, 2002.

VAN SCHOOTEN B.W., OP DEN AKKER R., ROSSET S., GALIBERT O., MAX A., ILLOUZ G., « Follow-up Question Handling in the IMIX and RITEL Systems: A Comparative Study », *Natural Language Engineering*, 1(1), pp. 1–23, 2007.

VILNAT A., Dialogue et analyse de phrases, mémoire d'Habilitation à Diriger des Recherches, University of Paris-Sud, Orsay, 2005.

VUURPIJL L.G., TEN BOSCH L., ROSSIGNOL S., NEUMANN A., PFLEGER N., ENGEL R., « Evaluation of Multimodal Dialog Systems », *Proceedings of the LREC Workshop on Multimodal Corpora and Evaluation*, Lisbon, Portugal, 2004.

WALKER M.A., « Can We Talk? Methods for Evaluation and Training of Spoken Dialogue Systems », *Journal of Language Resources and Evaluation*, 39(1), pp. 65–75, 2005.

WALKER M.A., PASSONNEAU R., BOLAND J.E., « Quantitative and Qualitative Evaluation of DARPA Communicator Spoken Dialogue Systems », *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, USA, pp. 515–522, 2001.

WALKER M.A., WHITTAKER S., STENT A., MALOOR P., MOORE J., JOHNSTON M., VASIREDDY G., « Generation and Evaluation of User Tailored Responses in Multimodal Dialogue », *Cognitive Science*, 28(5), pp. 811–840, 2004.

WARD N., TSUKAHARA W., « A Study in Responsiveness in Spoken Dialog », *International Journal of Human-Computer Studies*, 59(6), pp. 959–981, 2003.

WARREN M., *Features of Naturalness in Conversation*, John Benjamins, Amsterdam and Philadelphia, 2006.

WEIZENBAUM J., « ELIZA – A Computer Program For the Study of Natural Language Communication Between Man and Machine », *Communications of the Association for Computing Machinery*, 9(1), pp. 36–45, 1966.

WINOGRAD T., *Understanding Natural Language*, Academic Press, San Diego, 1972.

WRIGHT P., « Using Constraints and Reference in Task-oriented Dialogue », *Journal of Semantics*, 7, pp. 65–79, 1990.

WRIGHT-HASTIE H., POESIO M., ISARD S., « Automatically Predicting Dialogue Structure using Prosodic Features », *Speech Communication*, 36, pp. 63–79, 2002.

Index

V W

X Y Z