Multimodal Dialogue for Ambient Intelligence and Smart Environments

Ramón López-Cózar and Zoraida Callejas

1 Introduction

Ambient Intelligence (AmI) and Smart Environments (SmE) are based on three foundations: ubiquitous computing, ubiquitous communication and intelligent adaptive interfaces [41]. This type of systems consists of a series of interconnected computing and sensing devices which surround the user pervasively in his environment and are invisible to him, providing a service that is dynamically adapted to the interaction context, so that users can naturally interact with the system and thus perceive it as intelligent.

To ensure such a natural and intelligent interaction, it is necessary to provide an effective, easy, safe and transparent interaction between the user and the system. With this objective, as an attempt to enhance and ease human-to-computer interaction, in the last years there has been an increasing interest in simulating human-to-human communication, employing the so-called *multimodal dialogue systems* [46]. These systems go beyond both the desktop metaphor and the traditional speech-only interfaces by incorporating several communication modalities, such as speech, gaze, gestures or facial expressions.

Multimodal dialogue systems offer several advantages. Firstly, they can make use of automatic recognition techniques to sense the environment allowing the user to employ different input modalities, some of these technologies are automatic speech recognition [62], natural language processing [12], face location and tracking [77], gaze tracking [58], lipreading recognition [13], gesture recognition [39], and handwriting recognition [78].

Ramón López-Cózar
Dept. of Languages and Computer Systems, Faculty of Computer Science and Telecommunications, University of Granada, Spain, e-mail: rlopezc@ugr.es

Zoraida Callejas
Dept. of Languages and Computer Systems, Faculty of Computer Science and Telecommunications, University of Granada, Spain, e-mail: zoraida@ugr.es

H. Nakashima et al. (eds.), *Handbook of Ambient Intelligence and Smart Environments*,
DOI 10.1007/978-0-387-93808-0_21,

Secondly, these systems typically employ several output modalities to interact with the user, which allows to stimulate several of his senses simultaneously, and thus enhance the understanding of the messages generated by the system. These modalities are implemented using technologies such as graphic generation, natural language generation [33], speech synthesis [17], sound generation [24], and tactile/haptic generation [70].

Thirdly, the combination of modalities in the input and output allows to obtain more meaningful and reliable interpretations of the interaction context. This is, on the one hand, because complementary input modalities provide with non-redundant information which helps creating a richer model of the interaction. On the other hand, redundant input modalities increase the accuracy and reduce the uncertainty of the information [11]. Besides, both the system and the user can choose the adequate interaction modalities to carry out the communication, thus enabling a better adaptation to environmental conditions such as light/acoustic conditions or privacy. Furthermore, the possibility to choose alternative ways of providing and receiving information allows disabled people to communicate with this type of system using the interaction modalities that best suit their needs.

Researchers have developed multimodal dialogue systems for a number of applications, for example, interaction with mobile robots [44] and information retrieval [26]. These systems have also been applied to the main topic of this book, for example, to enhance the user-system interaction in homes [54, 23], academic centers [47], hospitals [8] and theme parks [56].

After this brief introduction, the remaining of this chapter is organised as follows. Section 2 deals with context awareness and user modelling. Section 3 addresses the handling of input and contextual information and how to build abstractions on the environment to acquire it. Section 4 centres on dialogue management, i.e. on how the system responds to inputs and to changes in the environment, focusing on interaction and confirmation strategies. Section 5 addresses response generation, discussing the *fission* of multimodal information by means of synthesised speech, sounds, graphical objects and animated agents. Section 6 addresses system evaluation, describing the peculiarities of the evaluation of multimodal interfaces for AmI and SmE applications. Section 7 presents the conclusions.

2 Context Awareness

Although there is not a complete agreement on the definition of *context*, the most widely accepted is the one proposed by [15]: "*Any information that can be used to characterize the situation of an entity (...) relevant to the interaction between a user and an application, including the user and the application themselves*". As can be observed from this definition, any information source can be considered context as long as it provides knowledge relevant to handle the communication between the user and the system. In addition, the user is also considered to be part of the contextual information.

Kang et al [38] differentiate two types of context: *internal* and *external*. The former describes the user state (e.g. communication context and emotional state), whereas the latter refers to the environment state (e.g. location and temporal context). Most studies in the literature focus on the external context. However, although external information, such as location, can be a good indicator of the user intentions in some domains, in many applications it is necessary to take into account more complex information sources about the user state, such as emotional status [9] or social information [48].

External and internal context are intimately related. One of the most representative examples is *service discovery*. As described in [16], service context (i.e. location of services and providers) can be matched against the user context as a filtering mechanism to provide only the services that are suitable given the user current state. Another example are the so-called *proactive systems*, which decide not only the information that must be provided to the user, but also when to do it to avoid interrupting. The factors that must be taken into account to decide in which cases the user should be interrupted comprise both the user context (e.g. the activities in which he is involved, his emotional state, social engagement or social expectations) and the context related to the environment, such as the location of the user and the utility of the message [32]. For example, [69] presents a proactive shopping assistant integrated in supermarket trolleys. The system observes the shopper's actions and tries to infer his goals. With this information, it proactivelly offers adapted support tailored to the current context, for example displaying information about products when the user holds them for a very long time or comparing different products when the user is deciding between two items.

2.1 Context Detection and Processing

Context information can be gathered from a wide variety of sources, which produces heterogeneity in terms of quality and persistence [30]. As described in [31], *static* context deals with invariant features, whereas *dynamic* context is able to cope with information that changes. The frequency of such changes is very variable and can deeply influence the way in which context is obtained. It is reasonable to obtain largely static context directly from users, and frequently changing context from indirect means such as sensors.

Once the context information has been obtained, it must be internally represented within the dialogue system so that it can be handled in combination with the information about the user interaction and the environment. In dynamic environments, the information describing context is highly variable and sometimes very unreliable and contradictory. Besides, it is also important to keep track of previous context during the interactions. This is why it is necessary to build consistent context models.

A number of methods have been proposed to create these models[1]. The simplest approach for modelling context is the key-value method, in which a variable (key) contains the actual context (value). However, key-values approaches are not suitable for capturing sophisticated contextual information. To solve this drawback, tagged encoding can be employed. It is based on context profiles using a markup scheme, which enables modelling and processing context recursively, and to employ efficient context retrieval algorithms. However, this approach still lacks capability for representing complex contextual relationships and sometimes the number of contextual aspects that can be described is limited. It was adopted for example in the Stick-e Notes system, which was based on the SGML standard language [59].

More sophisticated mechanisms are object oriented models (e.g. in [7]), which have the benefits of encapsulation and reusability. With them, it is possible to manage a great variety and complexity of contextual information while maintaining scalability. Following the philosophy of encapsulation, Kwon and Sadeh [42] presented a multi-agent architecture in which a negotiator agent encapsulates all the knowledge about context. Recently, Henricksen and Indulska [30] have presented a context modelling methodology which takes into account different levels of abstraction to cover the spectrum between the sensor output and the abstract representation that is useful for the application. This is an improvement over the first toolkits developed to represent context information, such as "The context toolkit" [14] which allowed the creation of abstract objects that could be related to the output of sensors. In addition, graphical representations [16] and ontology-based modelling are currently being employed for pervasive and mobile applications [25].

The context representation can be stored either locally, in a computer network, or in both. For example, the context-awareness system developed in the CASY project [79] enables users to define new *context events*, which can be sent along with multimedia messages through Internet. For example, a child can produce a context event "I am going to bed" and receive a "goodnight message" video which his grandmother had previously recorded, thus allowing geographically distributed families to communicate asynchronously.

The processing of context is essential in dialogue systems to cope with the ambiguities derived from the use of natural language. For instance, it can be used to resolve anaphoric references (e.g. decide the noun referred by a pronoun). However, the linguistic context sometimes is not enough to resolve a reference or disambiguate a sentence, and additional context sources must be considered. Porzel and Gurevych [60] identified four sources, namely situational, discourse, interlocutor and domain. These sources were employed in a mobile tourist information system to distinguish different types of *where* clauses. For example, for the user question "Where is the castle?" the system could answer two types of response: a path to the castle, or a description of the setting. The system selected the appropriate answer considering the proximity of the user to the castle, the intentions of the user (whether he wanted to go there or view it), and the user preferences (whether he wanted the shortest, fastest or nicest path).

[1] A comprehensive review of context modelling approaches can be found in [71].

Other important topics concerned with the processing of context are: relevant data structures and algorithms to be used, how to reduce the overhead caused by taking context into account, and possible recovery strategies to be employed when the environment is not providing the necessary services [67]. Additionally, privacy issues are gaining more attention from the scientific community [29].

2.2 User Modelling

As previously discussed, the *user* is also considered to be part of the *context*. Hence, it is very important to employ user models in AmI and SmE applications to differentiate user types and adapt the performance of the dialogue system accordingly. Besides, user models are essential for conversational interactions, as they enable to deal with interpersonal variations which can vary from a low level (e.g. acoustic features) to a higher level, such as features concerned with syntactic, vocabulary, pragmatic and speaking style. For example, these models enable to enhance speech recognition by selecting the most appropriate acoustic models for the current user. They are also very useful for speech generation, as system prompts can be created according to the level of experience of the user. For instance, pauses can be placed in specific points of the sentences to be synthesised in order to encourage experienced users to speak, even while the system is still talking (*barge-in*) [45].

The adaptation is specially useful for users of devices such as mobile phones, PDAs or cars, as they can benefit from the functionality built over the knowledge about them as an individual user. For example, this is the case of the in-car system presented in [4], which adapts the spoken interaction to the user as he books a hotel room while driving a car. In these cases, simple user models as *user profiles* can be employed, which typically contain data concerned with user needs, preferences and/or abilities. For example, the profiles employed by the DS-UCAT system [47] contain students' personal data and preferences for the interaction with the system, such as language (English or Spanish), spoken modality (enabled/disabled), gender for the system voice (male/female) and acceptation of messages incoming from the environment (enabled/disabled). Considering these profiles, the dialogue manager of the system employs interaction strategies (to be discussed in Sect. 4) which are adaptive to the different user types. For example, system-directed initiative is used for inexperienced users, whereas mixed-initiative is employed for experienced users.

A drawback of user profiles is that they require time on the part of the users to fill-in the required data. An alternative is to classify users into preferences' groups, considering that sometimes the information about preferences can be common for different applications. In order to facilitate the sharing of this information, a cross-application data management framework must be used, such as the one presented in [40]. This framework has the ability to infer data, so that applications can use *cross-profile* data mining algorithms. For example, the system can find users that are similar to the one who is currently interacting with it and whose preferences might coincide.

General user profiles are suitable to classify habits and preferences of the users in order to respond with the appropriate actions. However in some application domains it is also necessary to model more complex and variable parameters such as the user's intentions. In this case, an alternative to using general user profiles is to employ the so-called *online user modelling*, which can be defined as the ability of a system to model users at run-time [18]. These have been used for example in health promotion dialogue systems, in which it is necessary to model the attitude of the users towards following the therapist advices (e.g. whether the user is confident or demoralised) [64].

3 Handling of Input Information

In AmI and SmE applications, devices such as lamps, TV, washing machine or oven, and sensors such as microphones, cameras or RFID readers, generate information which is the input to the dialogue system. To enable the interaction between the user and a multimodal dialogue system, as well as the interaction between the system and the devices in the environment, all the generated information must be properly represented so that it can be handled to produce the system responses.

3.1 Abstracting the Environment

To implement the interaction between the dialogue system and the environment, the former typically accesses a *middleware* layer that represents characteristics of the environment, e.g. interfaces and status of devices. Using this layer, the dialogue system does not need to interact directly with the devices in the physical world, thus getting rid of their specific peculiarities. The implementation details of the entities in the middleware are hidden to the system, and thus it uses the same standard communication procedure for any entity in the environment. Moreover, this layer eases the setting up of dynamic environments where devices can be added or removed at run-time. The communication between the entities in the middleware and the corresponding physical devices can be implemented using standard access and control mechanisms, such as EIB (European Installation Bus) [54, 55] or SNMP (Simple Network Management Protocol) [49].

Researchers have implemented this layer in different ways. For example, Encarnaçao and Kirste [19] proposed a middleware called SodaPop (Self-Organizing Data-flow Architectures suPorting Ontology-based problem decomPosition), which supports the self-organisation of devices into appliance ensembles. To do this, their proposal considers goal-based interaction of devices, and distributed event processing pipelines.

Sachetti et al [65] proposed a middleware called WSAMI (Web Services for Ambient Intelligence) to support seamless access to mobile services for the mobile user,

either pedestrian, in a car or in a public transport. This middleware builds on the Web services architecture, whose pervasiveness enables services availability in most environments. In addition, the proposal deals with the dynamic composition of applications in a way that integrates services deployed on mobile, wireless and on the Internet. The authors developed three key WSAMI-compliant Web services to offer intelligent-aware multimodal interfaces to mobile users: i) linguistic analysis and dialogue management, ii) multimodality and adaptive speech recognition, and iii) context awareness.

Berre et al [5] proposed a middleware called SAMBA (Systems for AMBient intelligence enabled by Agents) to support the interaction and interoperability of various elements by encapsulating and representing them as *agents* acting as members on an Ambient IntelligenceAmbient Intelligence Elements Society, and by using *executable models* at run-time in support of interoperatibility.

Montoro et al [54] implemented a middleware using a blackboard [20] created from the parsing of an XML document that represents ontological information about the environment. To access or change the information in the blackboard, applications and interfaces employ a simple communication mechanism using XML-compliant messages which are delivered via HTTP. An interesting feature of this proposal is that it allows to attach linguistic information to each entity in the middleware in order to automatically create a spoken interface associated with the corresponding device. Basically, it is comprised of a *verb part* (actions that can be performed with the device), an *object part* (name that can be given to the device), a *modifier part* (that refers to the kind of entity), a *location part* (that refers to the location of the device in the environment), and an *additional information part* (which contains additional information about the entity in the middleware).

3.2 Processing Multimodal Input

A number of formalisms have been defined to represent the information of the user interaction captured by the sensors in the environment, for example, typed feature structures [10] and XML-based languages such as M3L (Multimodal Markup Language) [72] or MMIL (Multimodal Interface Language) [63, 65].

Many multimodal dialogue systems typically employ the so-called *frames*, which can be defined as abstract representations of real-world entities and their relationships. Each entity if constituted of a set of attributes called *slots*, which are filled with data extracted by the sensors. The filling of the slots can be carried out either incrementally or by means of heritage relationships between frames. Slots can contain a diversity of data types, for example, words provided by a speech recogniser [54, 66], screen coordinates provided by a electronic pend, and location information obtained via RFID [55, 56] or GPS. They can also contain *time stamps* regarding the moment at which the frames are created.

Frames can be classified into three types: input, integration and output frames. *Input* frames are employed to store the information generated by the devices or

captured by the sensors in the environment. Each input frame may contain empty slots if the obtained information is partial. *Integration* frames are created by the dialogue system as it handles the user interaction. The slots of these frames are filled by the combination of the slots in the input frames. The *output* frames are used to generate system responses (as will be discussed in Sect. 5). The dialogue system can use these frames to change the status of devices in the environment (e.g. turn on a lamp), or to transmit information to the user employing e.g. synthesised speech and graphics on a PDA screen. For example, [56] used frames to represent contextual information in a theme park application, where users move 3D puzzle pieces to assemble a car. Nigay and Coutaz [57] proposed to use a formalism similar to frames called *melting pots*, which encapsulate multimodal information with time stamps. These stamps are very important to consider the information provided by devices and sensors either as *complementary* or *independent*.

Taking into account temporal information, the combination (or *fusion*) of the information captured from the user can be either microtemporal, macrotemporal or contextual. The microtemporal fusion is carried out when the information captured from the user, e.g. stored in melting pots, is complementary and overlapped in time. The macrotemporal fusion is carried out when the information is complementary and the time intervals of the captured information do not overlap in time, but belong to the same analysis window. Finally, the contextual fusion combines information without considering time restrictions.

The fusion of the information can be carried out at *signal* or *semantic* levels. In the former case, only temporal issues are involved (e.g. data synchronisation) without taking into consideration any kind of high-level interpretation. This fusion type is usually employed for audiovisual speech recognition, which combines acoustic and lip movement signals [51]. On the contrary, the fusion at the semantic level deals with the combination of multimodal information interpreted separately, and then combined considering their meaning. Among others, Sachetti et al [65] employed fusion at the semantic level to combine information obtained from speech and 2-D gestures captured by an electronic pen.

4 Dialogue Management

Using the information generated by the devices, captured by the sensors in the environment and provided by the context, the dialogue system manages the interaction with the user and provides services. As the dialogue between the system and the user is multimodal, the dialogue manager controls the execution flow from various input and output communication channels, e.g. speech and/or icon selection on a PDA screen. The modalities to be used may be determined by the user profile and the contextual information about the environment conditions. If speech is suitable as an input modality, the user voice is captured by a microphone and is the input to a speech recogniser [54, 23, 66] which provides one or more recognition hypotheses. If speech is enabled as an output modality, the system can generate spoken

responses via synthesised speech, typically using text-to-speech (TTS) conversion [17]. For example, a spoken dialogue[2] between the user and the GENIO system [23], developed for home automation and entertainment, is as follows:

(1) *User*: Majordomo.
(2) *System*: Tell me.
(3) *User*: Wash my cloths by the time when I come back home.
(4) *System*: When are you coming back?
(5) *User*: At eight in the evening.
(6) *System*: Eight in the morning or in the evening?
(7) *User*: In the evening, I said in the evening.
(8) *System*: OK, eight in the evening. Which washing program?
(9) *User*: Soft, 30 degrees centigrade.
(10) *System*: Did you say 13 degrees?
(11) *User*: No, I said 30 degrees.
(12) *System*: OK, washing machine programmed for soft, 30 degrees centigrade. Anything else?
(13) *User*: Is the dishwasher working at the moment?
(14) *System*: No, it is not.

To enable this dialogue, the system employs predefined dialogue management strategies to decide what to do after processing the information obtained from the user, the environment and the context. Typical actions are: provide information to the user extracted from a database query, check or change the status of devices in the environment, confirm data, and ask for missing data necessary to perform a specific action.

Information can be missing either because a data item has not been provided by the user, or due to processing errors. For instance, a processing error can be observed in the dialogue shown above, as the data item 'in the evening' provided by the user in turn (5) was not understood by the system. This might happen because the words 'in the evening' uttered by the user were not recognised, or were recognised with very low *confidence score* (as will be discussed in Sect. 4.2) and thus were rejected by the system. As a result, the system needed to employ an additional dialogue turn (6) to clarify the time, thus making the dialogue less efficient.

In order to enhance the system performance and provide a friendly interaction, the dialogue management strategies must be adaptive to user characteristics such as knowledge, expertise, preferences and needs, which are stored in the user profile [27] [21]. Specifically, the adaptation to special needs is receiving much attention in terms of how to make systems usable by handicapped people [28], children [1] and the elderly [43]. Despite their complexity, these characteristics are to some extent rather static. Jokinen [37] identified another degree of adaptation in which the dialogue management strategies are not only adapted to the explicit message conveyed during the interaction, but also to the user's intentions and state. Following

[2] For illustration purposes, this dialogue has been slightly changed from the one shown in [23].

this guideline, affective computing focuses on how to recognize and dynamically adapt the conversation with the system to the user emotional state. For example, many breakdowns in man-machine communication could be avoided if the machine was able to recognize the emotional state of the user and responded to it more sensitively [50]. Earlier experiments showed that an empathetic computer agent can indeed contribute to a more positive perception of the interaction [61]. This study also examined how social factors, such as status, influence the semantic content, the syntactic form and the acoustic realization of conversations [74].

4.1 Interaction Strategies

To implement the user-system interaction by means of a multimodal dialogue, the system developers must design interaction strategies to control the way in which user and system interact with each other employing dialogue turns. For example, one issue to decide is whether the user must provide just the data items requested by the system, and in the order specified by the system, or alternatively, he is allowed to provide the data in the order that he wishes and be over-informative (provide data not requested by the system).

Another important issue to decide is who (user or system) has the initiative in the dialogue. In accordance with this criterion, three types of interaction strategies are often distinguished in the literature: *user-directed*, *system-directed* and *mixed*. When the former strategy is used, the user has always the initiative in the dialogue, and the system just responds his queries and orders. This is the case in the sample dialogue shown above. The main problem with this strategy is that the user may think that he is free to say whatever he wants, which tends to cause speech recognition and understanding errors.

When the *system-directed* strategy is used, the system has always the initiative in the dialogue, and the user just answers its queries. The advantage of this strategy is that it tends to reduce the possible user input, which typically leads to very efficient dialogues. The disadvantage is the lack of flexibility on the part of the user, who is restricted to behave as the system expects, providing the necessary data to perform some action in the order specified by the system.

When the *mixed* interaction strategy is used, both the user and the system can take the initiative in the dialogue, consequently influencing the conversation flow. The advantage is that the system can guide the user in the tasks that he can perform. Moreover, the user can take the initiative and be over-informative.

4.2 Confirmation strategies

Considering the limitations of the state-of-the-art recognition technologies employed to build multimodal dialogue systems, it is necessary to assume that the

information captured by sensors about the user interaction may be uncertain or ambiguous.

To deal with the uncertainty problem, these systems typically employ *confidence scores* attached to the frame slots, for example, real numbers in the range between 0 and 1. A confidence score under a threshold indicates that the data item in the slot must be either confirmed or rejected by the system. Two types of confirmation strategies are are often employed: *explicit* and *implicit*. When the former is used, the system generates an additional dialogue turn to confirm the data item obtained from the previous user turn. Turn (10) in the sample dialogue above shows an example of this confirmation type. The disadvantage of this confirmation method is that the dialogue tends to be lengthy due to the additional confirmation turns, which makes the interaction less effective and even excessively repetitive if all the data items provided by the user must be confirmed.

When the implicit confirmation strategy is used, no additional turns are necessary since the system includes the data items to be confirmed in the next dialogue turn to get other data items from the user. In this case, it is responsibility of the user to make a correction if he observes a data item wrongly obtained by the system. Turn (12) in the sample dialogue above shows an example of this confirmation strategy, where the system tries to get a confirmation for the data item 'Soft washing programme'. As the user did not make any correction in his next dialogue turn (13), the system assumed that this data item was confirmed.

The input information can be ambiguous if it can be interpreted in different ways by the system. For example, the input made with a pen on a touch-sensitive screen can refer to three different purposes: pointing (as a substitute of mouse), handwriting and drawing. In order to face this problem, the dialogue system must employ some method to try to automatically decide the mode in which the pen is being used, and/or employ an additional dialogue turn to get a confirmation from the user about the intended mode.

The strategies discussed above are useful to avoid misunderstandings and deal with the uncertainty of data obtained from the user. One related, but different situation is the non-understanding, which occurs when the system does not get any data from the user interaction. In this case, two typical strategies for handling the error are to ask the user to repeat the input, and to ask him to rephrase the sentence (in the case of spoken interaction).

5 Response Generation

After the analysis of the input information captured from the user and obtained from the environment, the dialogue system must generate a response for the user taking into account the context information and the user profile. To do this, it can employ the *output* frames discussed in Sect. 3.2. These frames are analysed by a module of the system that carries out the so-called *fission* of multimodal information, i.e., a decision about the devices and interaction modalities to be used for generating the

response. For example, the system can generate a multimodal response using synthesised speech, sounds, graphical objects and an animated agent in a PDA screen. The use of several modalities to generate system responses allows a friendlier interaction that better adapts to the user preferences and needs. Moreover, it enables the user to create a proper mental model of the performance of the system and the dialogue status, which facilitates the interaction and decreases the number of interaction errors. The response generation may involve as well to access the middleware to change the status of a device in the environment, e.g. to switch on a specific lamp in the living room [54].

Speech synthesis is typically carried out employing a text-to-speech (TTS) conversion [17], which transforms into speech any sentence in text format. The sentence can be created employing natural language generation techniques, which typically carry out two sequential phases, known as *deep* and *syntactic* (or *surface*) generation [33]. The deep generation can be divided into two phases: *selection* and *planning*. In the selection phase the system selects the information to be provided to the user (typically obtained from a database query), while in the planning phase it organises the information into phrase-like structures to achieve clarity and avoid redundancy. An important issue of this phase is the lexical selection, i.e. choosing the most appropriate words for each sentence. The lexical selection depends on the information previously provided by the system, the information available in the context, and specific stylistic considerations. The *syntactic* (or *surface*) generation takes the structures created by the deep generation and builds the sentences expressed in natural language, ensuring the grammatical correction.

In addition to speech, the system may generate sounds to provide a friendly interaction, for example, using *auditory icons* [24], which are associations between everyday sounds to particular events of the dialogue system.

The display of the communication device employed by the user may include graphical objects concerned with the task carried out by the system, e.g. pictures of a microwave oven, washing machine, dishwasher or lamps [55]. Additionally, the graphical objects can be used to provide visual feedback about the key data obtained form the user, for example, the temperature for the washing programme.

The display can also include the so-called *animated agents*, which are computer-animated characters, typically with the appearance of a human being, that produce facial expressions (e.g. by lips, eyes or eyebrows movements) and body movements in synchronisation with the synthesised speech. The goal of these agents is to provide a friendlier interaction, and ease the understanding of the messages generated by the system. For instance, the GENIO system [23] developed for home automation and entertainment, employs one of these agents to show a human-like majordomo on a PDA screen. Another sample animated agent is the robotic interface called *iCat* (Interactive Cat) [66], which is connected to an in-home network to control devices (e.g. light, VCR, TV, radio) and to access the Internet.

6 Evaluation

There are no generally accepted criteria for evaluating AmI and SmE systems. As it is a highly multidisciplinary area, authors usually employ standard methods to evaluate particular aspects of each system, rather than evaluating the system as a whole. This is the case of the evaluation of multimodal interfaces, which are usually assessed by employing the standard performance and satisfaction metrics of traditional dialogue systems [18].

Furthermore, the implementation of these systems is usually very costly. Therefore, regardless of the approach employed, it is desirable to assess their usability at design time, previously to implementation. Wizard of Oz techniques [22, 76] are typically employed to do this. Using these techniques, a human plays the role of some functionalities of the system and the users are made to believe that they are actually interacting with a machine. Once the validity of the design is checked, the next step is to create a prototype system to evaluate the system performance in real conditions. The results obtained from this evaluation are very useful to find out weaknesses of the system that must be improved before it is made available for the users.

6.1 Contextual Evaluation

As argued in [68], most context-aware systems provide a functionality which is only available or useful in a certain context. Thus, the evaluation must be carried out in a given context as well. In this way, there is more control over the data obtained, given that the situation is restricted. However, the evaluation results might not be significant to cope with the full spectrum of natural interactions. Also, simulating the type of data that would be acquired from the sensor system in a real scenario may be difficult.

As an attempt to solve these problems, the so-called *smart rooms* are usually employed. However, these environments fail to reproduce the variety of behaviours, movements and tasks that the users would perform on a daily-live basis, specially when these rooms are located in laboratories. An alternative are the so-called *Living labs*, i.e., houses built with ubiquitous sensing capabilities which provide naturalistic user behaviours in real life. However, as the user must interact in an environment which is not his real home, his behaviour may still be altered. In any case, user behaviour in living labs is still more natural than the one that could be obtained from short laboratory visits [35].

An example of living lab is the PlaceLab [34], a real house located near the MIT campus in a residential area, occupied by volunteers who can live there for periods of days, weeks and up to months depending of the studies being made. It was designed to be able to sense the opening and closing of doors and windows, electrical current, water and gas flow; and is equipped with cameras and microphones to track people,

so that routine activities of everyday home life could be observed, recorded and manipulated.

In the case of mobile AmI or SmE systems, the evaluation requires to consider several contexts in order to study the system response in different geographical locations. For example, the MATCH system [36], which provides multimodal access to city help in hand-held devices, was evaluated both in laboratory and mobile settings while walking in the city of New York. Another example is the exercise counselor presented in [6], which is a mobile conversational agent played on a PDA who motivates the users to walk more. It was evaluated both in laboratory and in two open conditions; firstly, with the test user walking on a treadmill while wearing the PDA, and secondly with the user wearing the PDA continuously during eight days.

6.2 Evaluation from the Perspective of Human-computer Interaction

It is very important to minimize the number of variables that influence the evaluation of systems in order to obtain representative results. An approach to achieve this goal is to employ a *single domain focus* [68]. That is, evaluating the system from a single perspective, keeping everything else constant. A typical usage of this approach is to evaluate the system from the perspective of the human-computer interaction. This is the case of the evaluation of some context-aware dialogue systems, such as SmartKom [73] and INSPIRE [52].

The SmartKom project provides an intelligent, multimodal human-computer interface for three main scenarios. The first is Smarktom Home/Office, which allows communication and operation of home appliances. The second is SmartKom Public, which allows access to public services. The third is SmartKom mobile, which is a mobile assistant. The authors differentiate between what developers and users need to evaluate the system [3]. For the developers, the objective of evaluation is to deliver reliable results of the performance under realistic application conditions, whereas from the users perspective the goal is to provide different services retrieved in a multimodal manner. An outcome of this project was a new framework specifically devoted to the evaluation of multimodal systems called PROMISE (Procedure for Multimodal Interactive System Evaluation) [2, 3]. This new framework is an extended version of a widely adopted framework for the evaluation of spoken dialogue systems, called PARADISE [75]. The main idea behind PROMISE is to define *quality* and *quantity* measures that can be computed during the system processing. Quality measures include system cooperativity, semantics, helps, recognition (speech, gestures and facial expressions), transaction success and ways of interaction. Quantity measures include barge-in, elapsed time, and users/system turns. A distinct aspect of PROMISE, not considered in PARADISE, is a weightening scheme for different recognition components, which allows solving contradictory inputs from the different modalities.

The PROMISE framework focuses mainly on the calculation of "objective" evaluation measures. However, in order to obtain a meaningful interpretation of user satisfaction, also "subjective" measures about perceived quality of the system should be gathered. Möller et al [53] identified some of the most important factors that influence the perceived quality of the service provided by a dialogue interface: *user*, *agent*, *environment*, *task* and *contextual* factors. User factors include attitude, emotions, experiences and knowledge which affect the users judgments. Agent factors are related to the characteristics of the machine counterpart. Environment factors relate to the physical context of the interaction. Task and contextual factors relate to the non-physical context. The INSPIRE system, designed to provide a multimodal interface for home, office and in-car conversational interactions, was evaluated taking into account these factors to assess the impact of the speech output component on the quality judgments of the users.

7 Conclusions

This chapter has presented a review of some important issues concerned with the design, implementation, performance and evaluation of multimodal dialogue systems for AmI and SmE environments. It has discussed the conceptual components of these systems, which deal with context awareness, multimodal input processing, dialogue management, and response generation.

The first part of the chapter has described context awareness approaches. It has introduced different context types (internal and external) and discussed context detection and processing. As the user is considered to be part of the context, the chapter has also focused on user modelling, discussing user profiles and on-line user modelling.

Next, the chapter has addressed the handling of input information. On the one hand, has taken into account the representation of the environment, which is typically modelled by a *middleware* layer, discussing different alternatives existing in literature to implement this. On the other hand, the chapter has discussed formalisms typically employed by the research community to represent multimodal input information. It has also addressed the *fusion* types employed to combine the internal representations of the multimodal information captured from the user or generated by the environment.

The third part of the chapter has addressed dialogue management, which enables the intelligent, adaptive, and natural multimodal interaction between the user and the environment. The chapter has discussed the kind of such strategies employed to decide the system responses and actions, taking into account predefined dialogue management strategies and user models. These approaches have been further discussed in terms of dialogue initiative and confirmation strategies.

The chapter has focused as well on response generation, discussing the procedure called *fission* of multimodal information, which is a decision about the devices and interaction modalities to be used for generating a system's response.

The last section of the chapter has addressed evaluation. It has initially discussed standard methods for the evaluation of dialogue systems, such as Wizard of Oz and prototyping. Then, it has addressed scenarios for the contextual evaluation of AmI and SmE applications, such as smart rooms and living labs. Finally, it has discussed evaluation from the perspective of human-computer interaction, focusing on the proposals of the SmartKom and INSPIRE projects.

Acknowledgements This work has been funded by the research project HADA - Adaptive Hypermedia for the Attention to the Diversity in Ambient Intelligence Environments (TIN2007-64718), funded by the Spanish Ministry of Education and Science.

References

[1] Batliner A, Hacker C, Steidl S, Nöth E, D'Arcy S, Russel M, Wong M (2004) Towards multilingual speech recognition using data driven source/target acoustical units association. In: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), Montreal, Quebec, Canada, pp 521–524

[2] Beringer N, Karal U, Louka K, Schiel F, Türk U (2002) PROMISE A procedure for multimodal interactive system evaluation. In: Proc. of the LREC Workshop on Multimodal Resources and Multimodal Systems Evaluation, Las Palmas, Spain, pp 77–80

[3] Beringer N, Louka K, Penide-Lopez V, Türk U (2002) End-to-end evaluation of multimodal dialogue systems - can we transfer established methods? In: Proc. of the LREC Workshop on Multimodal Resources and Multimodal Systems Evaluation, Las Palmas, Spain, pp 558–563

[4] Bernsen N (2003) User modelling in the car. Lecture Notes in Artificial Intelligence pp 378–382

[5] Berre AJ, Marzo GD, Khadraoui D, Charoy F, Athanasopoulos G, Pantazoglou M, Morin JH, Moraitis P, Spanoudakis N (2007) SAMBA - an agent architecture for ambient intelligence elements interoperability. In: Proc. of Third International Conference on Interoperability of Enterprise Software and Applications, Funchal, Madeira, Portugal

[6] Bickmore T, Mauer D, Brown T (2008) Context awareness in a handheld exercise agent. Pervasive and Mobile Computing Doi:10.1016/j.pmcj.2008.05.004. In press

[7] Bouzy B, Cazenave T (1997) Using the object oriented paradigm to model context in computer Go. In: Proc. of Context'97, Rio, Brazil

[8] Bricon-Souf N, Newman CR (2007) Context awareness in health care: A review. International journal of medical informatics 76:2–12

[9] Callejas Z, López-Cózar R (2008) Influence of contextual information in emotion annotation for spoken dialogue systems. Speech Communication 50(5):416–433

[10] Carpenter R (1992) The logic of typed feature structures. Cambridge University Press, Cambridge, England
[11] Corradini A, Mehta M, Bernsen N, Martin J, Abrilian S (2003) Multimodal input fusion in human-computer interaction. In: Proc. of the NATO-ASI Conference on Data Fusion for Situation Monitoring, Incident Detection, Alert and Response Management, Yerevan, Armenia
[12] Dale R, Moisl H, Somers H (eds) (2000) Handbook of natural language processing. Dekker Publishers
[13] Daubias P, Deléglise P (2002) Lip-reading based on a fully automatic statistical model. In: Proc. of International Conference on Speech and Language Processing, Denver, Colorado, US, pp 209–212
[14] Dey A, Abowd G (1999) The context toolkit: Aiding the development of context-enabled applications. In: Proc. of the SIGCHI conference on Human factors in computing systems (CHI 99), Pittsburgh, Pennsylvania, US, pp 434–441
[15] Dey A, Abowd G (2000) Towards a better understanding of context and context-awareness. In: Proc. of the 2000 Conference on Human Factors in Computer Systems (CHI'00), pp 304–307
[16] Doulkeridis C, Vazirgiannis M (2008) CASD: Management of a context-aware service directory. Pervasive and mobile computing Doi:10.1016/j.pmcj.2008.05.001. In press
[17] Dutoit T (1996) An introduction to text-to-speech synthesis. Kluwer Academic Publishers
[18] Dybkjaer L, Bernsen N, Minker W (2004) Evaluation and usbility of multimodal spoken language dialogue systems. Speech Communication 43:33–54
[19] Encarnaçao J, Kirste T (2005) Ambient intelligence: Towards smart applicance ensembles. In: From Integrated Publication and Information Systems to Virtual Information and Knowledge Environments, pp 261–270
[20] Engelmore R, Mogan T (1988) Blackboard systems. Addison-Wesley
[21] Forbes-Riley K, Litman D (2004) Modelling user satisfaction and student learning in a spoken dialogue tutoring system with generic, tutoring, and user affect parameters. In: Proc. of the Human Language Technology Conference - North American chapter of the Association for Computational Linguistics annual meeting (HLT-NAACL'06), New York, US, pp 264–271
[22] Fraser M, Gilbert G (1991) Simulating speech systems. Computer Speech and Language 5:81–99
[23] Gárate A, Herrasti N, López A (2005) Genio: An ambient intelligence application in home automatation and entertainment environment. In: Proc. of Joint soc-EUSI Conference, pp 241–245
[24] Gaver WW (1992) Using and creating auditory icons. SFI studies in the sciences of complexity, Addison Wesley Longman, URL `Proceedings/1992/Gaver1992.pdf`
[25] Georgalas N, Ou S, Azmoodeh M, Yang K (2007) Towards a model-driven approach for ontology-based context-aware application development: a case study. In: Proc. of the fourth International Workshop on Model-Based Method-

ologies for Pervasive and Embedded Software (MOMPES '07), Braga, Portugal, pp 21–32
[26] Gustafson J, Bell L, Beskow J, Boye J, Carlson R, Edlund J, Granstrom B, House D, Wirén M (2000) Adapt - a multimodal conversational dialogue system in an apartment domain. In: Proc. of International Conference on Speech and Language Processing, Beijing, China, pp 134–137
[27] Haseel L, Hagen E (2005) Adaptation of an automotive dialogue system to users' expertise. In: Proc. of 9th International Conference on Spoken Language Processing (Interspeech'05-Eurospeech), Lisbon, Portugal, pp 222–226
[28] Heim J, Nilsson E, Havard J (2007) User Profiles for Adapting Speech Support in the Opera Web Browser to Disabled Users. Lecture Notes in Computer Science 4397:154–172
[29] Hengartner U, Steenkiste P (2006) Avoiding privacy violations caused by context-sensitive services. Pervasive and mobile computing 2:427–452
[30] Henricksen K, Indulska J (2006) Developing context-aware pervasive computing applications: models and approach. Pervasive and mobile computing 2:37–64
[31] Henricksen K, Indulska J, Rakotonirainy A (2002) Modeling context information in pervasive computing systems. In: Proc. of the First International Conference on Pervasive Computing, pp 167–180
[32] Ho J, Intille S (2005) Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In: Proc. of the 2005 Conference on Human Factors in Computer Systems (CHI'05), Portland, US, pp 909–918
[33] Hovy EH (1993) Automated discourse generation using discourse relations. Artificial Intelligence, Special Issue on Natural Language Processing 63:341–385
[34] Intille S, Larson K, Munguia E (2003) Designing and evaluating technology for independent aging in the home. In: Proc. of the International Conference on Aging, Disability and Independence
[35] Intille S, Larson K, Beaudin J, Nawyn J, Tapia EM, Kaushik P (2005) A living laboratory for the design and evaluation of ubiquitous computing technologies. In: Proc. of the 2005 Conference on Human Factors in Computer Systems (CHI'05), Portland, Oregon, US, pp 1941–1944
[36] Johnston M, Bangalore S, Vasireddy G, Stent A, Ehlen P, Walker M, Whittaker S, Maloor P (2002) Match: An architecture for multimodal dialogue systems. In: Proc. of Association for Computational Linguistics, Pennsylvania, Philadelphia, US, pp 376–383
[37] Jokinen K (2003) Natural interaction in spoken dialogue systems. In: Proc. of the Workshop Ontologies and Multilinguality in User Interfaces, Crete, Greece, pp 730–734
[38] Kang H, Suh E, Yoo K (2008) Packet-based context aware system to determine information system user's context. Expert systems with applications 35:286–300
[39] Kettebekov S, Sharma R (2000) Understanding gestures in multimodal human computer interaction. Int Journal on Artificial Intelligence Tools 9(2):205–223

[40] Korth A, Plumbaum T (2007) A framework for ubiquitous user modelling. In: Proc. of IEEE International Conference on Information Reuse and Integration, Las Vegas, Nevada, US, pp 291–297
[41] Kovács GL, Kopácsi S (2006) Some aspects of ambient intelligence. Acta Polytechnica Hungarica 3(1):35–60
[42] Kwon O, Sadeh N (2004) Applying case-based reasoning and multi-agent intelligent system to context-aware comparative shopping. Decision Support Systems 37:199–213
[43] Langner B, Black A (2005) Using speech in noise to improve understandability for elderly listeners. In: Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'05), San Juan, Puerto Rico, pp 392–396
[44] Lemon O, Bracy A, Gruenstein A, Peters S (2001) The WITAS Multi-Modal Dialogue System I. In: Proc. of Interspeech, Aalborg, Denmark, pp 1559–1562
[45] Levin E, Levin A (2006) Dialog design for user adaptation. In: Proc. of the International Conference on Acoustics Speech Processing, Toulouse, France, pp 57–60
[46] López-Cózar R, Araki M (2005) Spoken, Multilingual and Multimodal Dialogue Systems: Development and Assessment. John Wiley Sons
[47] López-Cózar R, Callejas Z, Montoro G (2006) DS-UCAT: A new multimodal dialogue system for an academic application. In: Proc. of Interspeech '06 - Satellite Workshop Dialogue on Dialogues, Multidisciplinary Evaluation of Advanced Speech-Based Interactive Systems, Pittsburgh, Pennsylvania, US, pp 47–50
[48] Markopoulos P, de Ruyter B, Privender S, van Breemen A (2005) Case study: bringing social intelligence into home dialogue systems. Interactions 12(4):37–44
[49] Martínez AE, Cabello R, Gómez FJ, Martínez J (2003) INTERACT-DM. A solution for the integration of domestic devices on network management platforms. In: Proc. of IFIP/IEEE International Symposium on Integrated Network Management, Colorado Springs, Colorado, US, pp 360–370
[50] Martinovski B, Traum D (2003) Breakdown in human-machine interaction: the error is the clue. In: Proc. of the ISCA Tutorial and Research Workshop on Error Handling in Dialogue Systems, Chateau d'Oex, Vaud, Switzerland, pp 11–16
[51] McAllister D, Rodman R, Bitzer D, Freeman A (1997) Lip synchronization of speech. In: Proc. of ESCA Workshop on Audio-Visual Speech Processing (AVSP'97), Kasteel Groenendael, Hilvarenbeek, The Netherlands, pp 133–136
[52] Möller S, Krebber J, Raake A, Smeele P, Rajman M, Melichar M, Pallotta V, Tsakou G, Kladis B, Vovos A, Hoonhout J, Schuchardt D, Fakotakis N, Ganchev T, Potamitis I (2004) INSPIRE: Evaluation of a Smart-Home System for Infotainment Management and Device Control. In: Proc. of the International Conference on Language Resources and Evaluation (LREC), Lisbon, Portugal, pp 1603–1606

[53] Möller S, Krebber J, Smeele P (2006) Evaluating the speech output component of a smart-home system. Speech Communication 48:1–27
[54] Montoro G, Alamán X, Haya P (2004) A plug and play spoken dialogue interface for smart environments. In: Proc. of Fifth International Conference on Intelligent Text Processing and Computational Linguistics (CICLing'04), Seoul, South Korea, pp 360–370
[55] Nazari AA (2005) A Generic UPnP Architecture for Ambient Intelligence Meeting Rooms and a Control Point allowing for Integrated 2D and 3D Interaction. In: Proc. of Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-Aware Services, USges and Technologies, pp 207–212
[56] Ndiaye A, Gebhard P, Kipp M, Klessen M, Schneider M, Wahlster W (2005) Ambient intelligence in edutainment: Tangible interaction with life-like exhibit guides. Lecture Notes in Artificial Intelligence 3814:104–113
[57] Nigay L, Coutaz J (1995) A generic platform for addressing the multimodal challenge. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems, ACM, Denver, Colorado, US, pp 98–105
[58] Ohno T, Mukawa N, Kawato S (2003) Just blink your eyes: A head-free gaze tracking system. In: Proc. of Computer-Human Interaction, Fort Lauderdale, Florida, pp 950–951
[59] Pascoe J (1997) The Stick-e note architecture: Extending the interface beyond the user. In: Proc. of the International Conference on Intelligent User Interfaces, Orlando, Florida, US, pp 261–264
[60] Porzel R, Gurevych I (2002) Towards context-sensitive utterance interpretation. In: Proc. of the 3rd SIGdial Workshop on Discourse and Dialogue, Philadelphia, US, pp 154–161
[61] Prendinger H, Mayer S, Mori J, Ishizuka M (2003) Persona effect revisited. using bio-signals to measure and reflect the impact of character-based interfaces. In: Proc. of the 4th International Working Conference on Intelligent Virtual Agents (IVA'03), Kloster Irsee, Germany, pp 283–291
[62] Rabiner LR, Juang BH (1993) Fundamentals of Speech Recognition. Prentice-Hall
[63] Reithinger N, Lauer C, Romary L (2002) MIAMM - Multidimensional information access using multiple modalities. In: Proc. of International CLASS workshop on natural intelligent and effective interaction in multimodal dialogue systems
[64] de Rosis F, Novielli N, Carofiglio V, Cavalluzzi A, de Carolis B (2006) User modeling and adaptation in health promotion dialogs with an animated character. Journal of Biomedical Informatics 39:514–531
[65] Sachetti D, Chibout R, Issarny V, Cerisara C, Landragin F (2004) Seamless access to mobile services for the mobile user. In: Proc. of IEEE Int. Conference on Software Engineering, Beijing, China, pp 801–804
[66] Saini P, de Ruyter B, Markopoulos P, Breemen AV (2005) Benefits of social intelligence in home dialogue systems. In: Proc. of 11th International Conference on Human-Computer Interaction, Las Vegas, Nevada, US, pp 510–521

[67] Satyanarayanan M (2002) Challenges in implementing a context-aware system. IEEE Distributed Systems Online 3(9)
[68] Schmidt A (2002) Ubiquitous computing - computing in context. PhD thesis, Lancaster University
[69] Schneider M (2004) Towards a Transparent Proactive User Interface for a Shopping Assistant. In: Proc. of Workshop on Multi-User and Ubiquitous User Interfaces (MU3I), Funchal, Madeira, Portugal, vol 3, pp 10–15
[70] Shimoga KB (1993) A survey of perceptual feedback issues in Dexterous telemanipulation: Part II. Finger Touch Feedback. In: Proc. of the IEEE Virtual Reality Annual International Symposium, Piscataway, NJ, IEEE Service Center
[71] Strang T, Linnhoff-popien C (2004) A context modeling survey. In: Proc. of Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004, Nottingham, England
[72] Wahlster W (2002) Smartkom: Fusion and fission of speech, gestures, and facial expressions. In: Proc. of First International Workshop on Man-Machine Symbiotic Systems, pp 213–225
[73] Wahlster W (ed) (2006) SmartKom: Foundations of Multimodal Dialogue Systems. Springer
[74] Walker M, Cahn J, Whittaker S (1997) Improvising linguistic style: Social and affective bases of agent personality. In: Proc. of the 1st International Conference on Autonomous Agents (Agents'97), Marina del Rey, CA, US, pp 96–105
[75] Walker M, Litman D, Kamm C, Abella A (1998) Evaluating spoken dialogue agents with PARADISE: Two case studies. Computer Speech and Language 12(4):317–347
[76] Whittaker S, Walker M (2005) Evaluating dialogue strategies in multimodal dialogue systems. In: Minker W, Buehler D, Dybkjaer L (eds) Spoken Multimodal Human-Computer Dialogue in Mobile Environments, Kluwer
[77] Yang J, Stiefelhagen R, Meier U, Waibel A (1998) Real-time face and facial feature tracking and applications. In: Proc. of Workshop on audio-visual speech processing, pp 79–84
[78] Yasuda H, Takahashi K, Matsumoto T (2000) A discrete HMM for online handwriting recognition. Pattern Recognition and Artificial Intelligence 14(5):675–689
[79] Zuckerman O, Maes P (2005) Awareness system for children in distributed families. In: Proc. of the 2005 International Conference on Interaction design for children (IDC 2005), Boulder, Colorado, US