Chapter 2
State of the Art

Many areas of artificial intelligence (AI) have had to struggle in various ways in which contextual dependencies arise, e.g. knowledge representation, natural language processing or expert systems to name a few. As the importance of context is frequently glossed over in the literature, researchers noted already in the early 80s that the denotation of the term has become murkier throughout its extensions in different fields of AI, calling it a *conceptual garbage can* [Clark and Carlson, 1981]. A classic AI example of contextual computing showed how the medicinal expert system MYCIN [Wallis and Shortliffe, 1982] can benefit from contextual considerations when prescribing treatments, resulting in fewer fatal intoxications as a result from the prescription [McCarthy, 1984]. This case constitutes a classic example as it establishes a blueprint for the so-called *representational* approaches to contextual computing in AI [Dourish, 2001, Dey, 2001]. Initially, we find an expert system that prescribes treatments based solely on the diagnosed disease. The subsequent contextual addition is - as will be discussed in greater detail below - twofold: firstly, a set of parameters is defined, i.e. which other medications the user is also taking, and secondly, a set of rules stating what to prescribe if certain parameter settings hold.

The following sections I will provide an initial discussion of context definitions and the role contextual computing has played in the targeted area of artificial intelligence, natural language processing and multimodal systems.

2.1 Defining Context

A general feature of context and contextual computing is lack of consensus concerning of the word itself. Over the past years many researches in computer science and other areas provided a vast and diverse number of definitions. This has prompted researchers to employ latent semantic analysis (LSA) and principal component analysis on a corpus of 150 definitions for context to find prominent similarities and divergences [Foltz et al., 1998, Bazire and Brézillon, 2005].

R. Porzel, *Contextual Computing: Models and Applications*, Cognitive Technologies,
DOI 10.1007/978-3-642-17396-7_2, © Springer-Verlag Berlin Heidelberg 2011

As both LSA and the subsequent clustering showed, the definitions were very diverse and, in general, dependent on the discipline in which they originated. Bazire and Brezillon (2005) extracted the following central components that are also shown in Figure 2.1: the user, the item, the environment, the observer and the context that influences them - as well as the relations between the context and the components and the relations among the components.

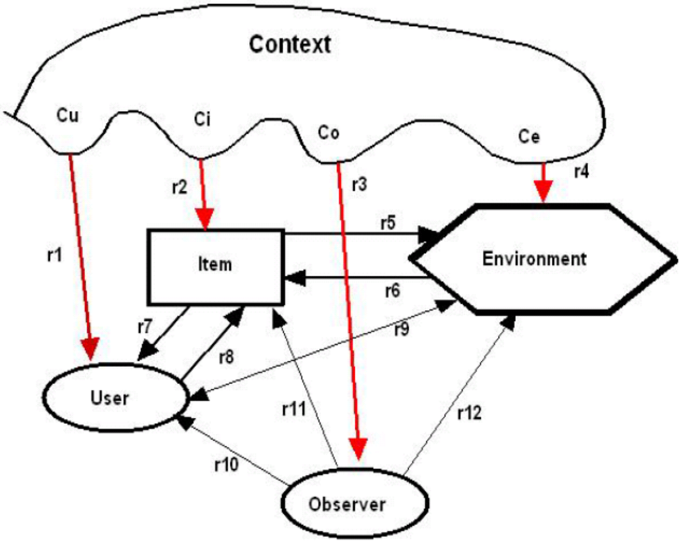

Fig. 2.1 Components and relations that appear in context definitions as extracted by Bazire and Brezillon (2005)

Their work shows that any definition highlights some subset of the components and relations. However, depending on the scientific area, each definition covers only a subset of the entire ensemble of components and relations and either omits or merges those components and relations that, given their own domain context, do not seem to matter. Below some sample definitions - that illustrate this domain-dependent diversity and selectivity - are listed:

- things or events related in a certain way [Ogden and Richards, 1923];
- paths of the information retrieval [Boy, 1991];
- a window on the screen [Abu-Hakima et al., 1993];
- a set of preferences or beliefs [Cahour and Karsenty, 1993];
- an infinite and partially known collection of assumptions [Turner, 1993].

In defining context for the domain of contextual computing individual definitions [Schilit et al., 1994, Dey, 2001] - as examples of the representational approach [Dourish, 2001] - constitute further instances of the model of Bazire and Brezillon (2005). For example, Dey's definition of 2001 highlights information that can be used to characterize entities, such as person, place, or object, as well as the user and the application (components in the model shown in Figure 2.1. Analogously, from the viewpoint of context-aware computing Schilit et al (1994) highlight the location of use, the collection of nearby people and dynamically changing objects as components to which context-aware system can adapt their interaction. The critical notion of relevancy is added in Dey's definition of a context-aware system, i.e., that *it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task*. The ensuing question concerning this determination of relevancy will be discussed shortly in Section 2.2, a further terminological discussion of context in this light is also provided by Dourish (2004) from an ethnomethodological perspective [Dourish, 2004].

Given these components of the Bazire and Brezillon (2005) model and the freely specifiable relations among them, also dictionary definitions can be seen in the of components and their relations. For example, the Merriam Webster Dictionary [Merriam-Webster, 2003] defines context as: *the interrelated conditions in which something exists or occurs*. According to the Oxford English Dictionary [Soanes and Stevenson, 2005], the term *context* usually has two primary senses:

1. the words around a word, phrase, statement, etc. often used to help explain (fix) the meaning;
2. the general conditions (circumstances) in which an event, action, etc. takes place.

Clearly, the first meaning is closely related to linguistic sense and the linguists' use of the term, whereas the second sense is the one which is closer to a desirable account of context in AI. This is also congruent to the observation by McCarthy (1986) who states that:

> [A]lmost all previous discussion of context has been in connection with natural language. However, I believe the main AI uses of formalized context will not be in connection with communication but in connection with reasoning about the effects of actions directed to achieving goals. It's just that natural language examples come to mind more readily.[McCarthy, 1986]

The definition of Angeles (1981) reflects the latter desideratum expressed by McCarthy more satisfactorily, as follows:

> context (L. contexere, *to weave together*. from con, 'with', and texere, 'to weave'): The sum total of meanings (associations, ideas, assumptions, preconceptions, etc.) that (a) are intimately related to a thing, (b) provide the origins for, and (c) influence our attitudes, perspectives, judgments, and knowledge of that thing.[Angeles, 1981]

Finally, a set of useful insights are presented in Collins Cobuild English Language Dictionary [Cobuild, 1995], which lists prevalent meanings of the term as follows:

1. The context of something consists of the ideas, situations, events, or information that relate to it and make it possible to understand it fully.
2. If something is seen in context or if it is put into context, it is considered with all the factors that are related to it rather than just being considered on its own, so that it can be properly understood.

3. If a remark, statement, etc. is taken or quoted out of context, it is only considered on its own and the circumstances in which it was said are ignored. It, therefore, seems to mean something different from the meaning that was intended.

Let me refer back to the work of Bazire and Brezillon's (2005) more comprehensive analysis of context definitions summarized above at this point and conclude this section by reiterating the main points one can take home from looking at the various definitions of context:

- Given that meanings - whether one considers the meaning of a verbal or nonverbal action - always arise within (or interwoven with) a given context, it becomes clear that this meaning is lost - or harder to recover - when things are taken *out of context*; in most of the cases I will examine herein they become ambiguous or underspecified.
- Given the specific entity under scrutiny only a subset of all possible components and their relations with it are pertinent for constructing that entities meaning, which explains why specific definitions of context focus only on those components and relations they deem pertinent for that entity.

In the following I will briefly provide an overview of the consequent attempt to formalize context correspondingly.

2.2 Fleshing Out Context

The basic intuition behind explicating contextual dependencies was that any given axiomatization of a state of affairs, meanings or relations presupposes an implicit context. Any explicit context model employed in processing information should, therefore, provide the information why a particular meaning can be assigned to the information and applied to the processing. In the literature this approach has often been called *fleshing out* and is considered impossible in its maximal form:

> It is seen that for natural languages a fleshing-out strategy – converting everything into decontextualized eternal sentences – cannot be employed since one does not always have full and precise information about the relevant circumstances. [Akman and Surav, 1996]

Before examining context and contextual computing in the domain of natural language I will shortly introduce the influential notions of McCarthy on context in AI [McCarthy, 1977, McCarthy, 1986]. McCarthy (1977) states that there can never be a most general context in which all stated axioms hold and everything is meaningful. This means that whenever an axiom is written it holds true only within the implicit context assumed, and one can always find a different context in which the axiom fails. Thusly, he proposes a relativized-truth-within-a-context by stating that a given statement p **is t**rue - abbreviated as *ist* - only in a given context c, which he, consequently writes as:

$$ist(p,c)$$

This states that a formal statement, such as discussed in greater detail in Section 3.1.1, called p, holds in context c. The motivation behind this formalization lies in the increased scalability, as axioms holding in a restricted blocks-world can be *lifted* to more general contexts, e.g. for systems in which context can change. Secondly,one can define vocabularies that have context-specific meanings, as frequently found in natural language. However, while this provides the formal means to employ subsumption, or in McCarthy's terms to be able to transcend a context, it leaves open the question when to transcend and where to. Taking the viewpoint of corresponding frameworks for handling dynamic domains, e.g. situation calculus [McCarthy and Hayes, 1969], of McCarthy and Hayes one has to face the so-called *framing* problem, where - from the top-down perspective - one needs to specify when a pertinent change in the background of a frame should be evoked, because its effects the meaning of something of the foreground of the frame [Mccarthy, 1979]. In so-called *representational* approach to contextual computing, the ensuing challenge is to specify when contexts are lifted/descended or become changed in the background [Dourish, 2001].

Dourish (2001) points out that current implementations of context-dependency or context-awareness in computational systems follow an almost standardized path. Firstly, a set of possible environmental states of contextually relevant parameters are defined; then, rules are implemented that try to match sensory inputs to one of the given states during runtime.[1] Within these types of applications context-awareness is fundamentally provided by such matching processes and context itself is represented by the predefined and stored set of environmental settings.

The contributions of Dourish's work (2001) are to point out not only the difficulties of determining the appropriate settings or states of the pertinent parameters, but also that the fundamental problem of this approach to contextual computing hinges of the question of how one can pre-compile all the settings and parameters that may become pertinent in advance. In his mind it is quite impossible to define these settings and parameters based solely on past research, surveys, testing, own experience, and on the purpose of the particular system alone.

Especially in such versatile instruments as natural language it becomes virtually impossible to predict all the possible utterances and the corresponding contextual dependencies on which their interpretation might hinge. But even in seemingly less murky waters human behavior can hardly be predicted as pointed out frequently by the example of cell phone use. It can be observed that people use their mobile phone as a watch, and although one would assume that despite the common assumption that it would be uncomfortable to pull something out of one's pocket to see what time it is, however, the number of people wearing wrist-watches has decreased. A similar development affects the use of alarm clocks. Although originally intended as a "remembering function" this property is often used instead of a conventional alarm clock, especially when traveling. Lastly the employment of Short Message Services (SMS) has greatly surprised the designers of mobile phones. Originally

[1] This matching process commonly involves a thresholding based the measured parameters and in case of ambiguous results various mediation techniques are used in order to determine a contextual state [Dey and Mankoff, 2005].

intended as a means to relay system-related information the capacity of one message was designed to be quite limited. Despite of this limitation and a hardly intuitive interface for entering them, SMS has become an every-day way of communication among people. In order to cope with the limitation of the message length novel abbreviations have been negotiated and completely unanticipated new writing styles have emerged, e.g. the so-called *Camel Case* sentences, such as *HowAreYou*, that are found in written messages - as SMS - where spaces between letters cost as much as the letters themselves.

The examples mentioned above show that people may use and interact with technology in unexpected ways. This reveals a fundamental problem of implementing a predefined set of settings as such approaches will inevitably not scale to cover possible interactions and behavior that will occur or evolve in future. According to Dourish the reason for this problem is that context has been approached as a representational problem by assuming the following properties of context [Dourish, 2004]:

- context is a form of information, i.e. context is seen as something that can be known, represented and encoded in software systems;
- context is delineable, i.e. it is thought to be possible to define what counts as context for a specific application in advance,
- context is stable, i.e. while context may vary from application to application, it does not vary from instance to instance of an interaction with an application;
- context and activity is separable, i.e. context is taken to describe features of the environment within which an activity takes place but the elements of the activity do not belong to context itself.

I will return to these general questions concerning representational approaches to contextual computing throughout the following sections as well as in Section 5.3, but will now shift the focus to the domain of context as it relates to natural language processing and the study of human communication.

2.3 Context in Language

In linguistics the study of the relations between linguistic phenomena and aspects of the context of language use is called *pragmatics*. Any theoretical or computational model dealing with reference resolution, e.g. anaphora- or bridging resolution, spatial- or temporal deixis or non-literal meanings requires taking the properties of the context into account. In current knowledge-based spoken dialogue systems *contextual interpretation* follows *semantic interpretation* - where the result of morphosyntactic analysis of the automatic speech recognition (ASR) output, as depicted in Figure2.2, is mapped to logic statements [Allen, 1987].

Multimodal systems - to be discussed in greater detail below - additionally fuse the results of semantic interpretation with the results of the other modality-specific analyzers. That is, the modality-specific signals, (e.g. speech or gesture) are transferred into graphical representations (e.g. word- or gesture graphs) by means of the

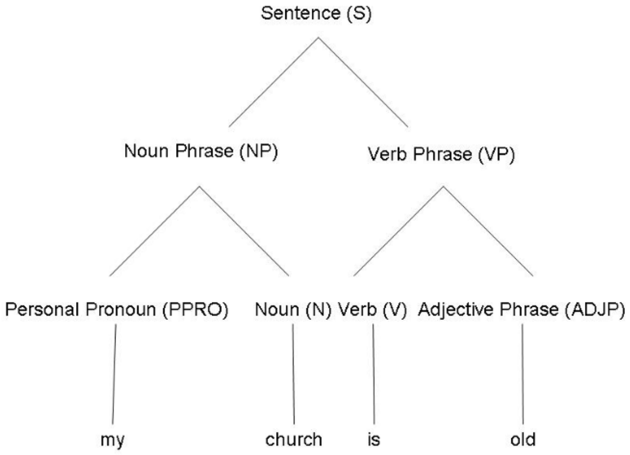

Fig. 2.2 A morpho-syntactic analysis of a set of words, showing that *my* is an instance of a *personal pronoun* and *church* is one of a *noun* and both together they act as a *noun phrase*

modality-specific recognizers, mapped onto their corresponding meaning representation and then fused using time-dependent unification techniques.

Contextual interpretation as described by Allen (1987) actually refers to the grounding of logical forms, e.g. of a logical statement expressing the proposition that my church is green. In grounding the form of a given logical statement, e.g. the referent of the referring expression *my church* in the utterance given in Figure 2.2, a corresponding instance of the form is determined. This, however, implies that context-independent graphical and semantic representations can be computed and the context-dependent contributions follow the semantic interpretation, resulting in a final *grounded* representations. I will provide a more detailed discussion of formal models for representing the meaning of an utterance in Sections 3.1.1 and 4.1.2.

This so-called *modular* view supports a distinct study of meaning (corresponding to the semantic representation) without having to muck around in the murky waters of language use. This view is supported by the claim that some semantic constraints seem to exist independent of context. In this work,I assume a different view that also allows for context-independent constraints, but offers a less modular point of view of contextual interpretation. I will show that contextual analysis can be employed already at the level of speech recognition, during semantic interpretation and, of course, thereafter. The central claim is being made, that - as in human processing - contextual information & knowledge can be used successfully in a computational framework in all processing stages.

In recent times the so-called *modular* theory of cognition [Fodor, 1983] has been abandoned more or less completely. The so-called *new look* or modern cognitivist positions hold that nearly all cognitive processes are interconnected, and freely exchange information; e.g. influences of semantic and pragmatic features have been shown to arise already at the level of phonological processing [Bergen, 2001]. While most research in linguistics, has consequently departed from this view, most computational approaches still feature a modular pipeline architecture in that respect.

In linguistics utterances which are context-dependent are called *indexical* utterances [Bunt, 2000]. Indexical utterances are - by virtue of the pervasiveness of contextual knowledge - the norm in discourse, with linguistic estimations of declarative non-indexical utterances around 10% [Barr-Hillel, 1954]. Without contextual knowledge utterances, or fragments thereof, become susceptible of interpretation in more than one way. Computer languages are designed to avoid anaphoric, syntactic, semantic and pragmatic ambiguity, but human languages seem to be riddled with situations where the listener has to choose between multiple interpretations. In these cases one says that the listener performs *pragmatic analysis*; corresponding to contextual interpretation on the computational side. For human beings the process of resolution is often unconscious, to the point that it is sometimes difficult even to recognize that there ever was any ambiguity.[2]

The phenomenon that this process of resolution, frequently goes unnoticed is due to the fact that in many cases the ambiguity is only perceived if the contextual information & pragmatic knowledge that allowed the listener interpret the utterance unambiguously are missing. These utterances/texts, therefore, become ambiguous only after they have been taken out of context, and, for example, appeared as a text(-fragment) in a linguistics textbook. The problem for computational linguistics originates - at least partially - in the fact language understanding has to make do with exactly such a contextually and pragmatically impoverished input.

Let us consider some examples and how they are treated in the literature, using the sample dialog shown in example 7.

(7) (a) User: OK, um, suppose I want to go to a museum tomorrow, which museum would you advise me?
(b) WoZ: You can visit the modern art museum.
(c) User: What is the exhibit, does it have like any architectural things inside there because I more like, you know, buildings and architectural things than, you know,

Setting up a referent is discourse is usually done by means of a referring expression (usually syntactically packaged as a noun phrase). As shown by Poesio about 50% of all noun phrases in their corpora are discourse-new, e.g. in an utterance such as shown in Example (7a) the referring expression *a museum* is considered non-anaphoric, i.e. discourse-new [Poesio and Vieira, 1998]. Anaphoric noun phrases make up 30% of their data, e.g. *it* in Example (7c) constitutes an anaphoric expression and is, hence, called the *anaphora*, which features a specific relation to

[2] Fauconnier and Turner (2002) name some potential reasons why this may be the case from an evolutionary perspective.

its antecedent (i.e. the referring expression *the museum*). This relation is termed *co-reference* as both forms denote the same referent, i.e. a specific museum. The remaining 20% of noun phrases are made up by so-called *associative* expressions, such as bridging expressions e.g. *the exhibit* in Example (7c) is considered a bridging expression, as the employment of the definite article is licensed by the fact that the speaker assumes that the interlocutor knows that museums feature exhibits. Human annotators can reliably mark (indefinite) discourse-new and anaphoric expressions, but reliability decreases for associative expressions and those cases where discourse-new referents are introduced by definite articles, due to common world knowledge, as in *The first man on the moon* [Poesio, 2002]. The problem arises as the borderline between these cases and bridging expressions is not very clear, causing the annotator inter-reliability to decrease.

Given the distinction made by Poesio (1998) most computational approaches have focused on resolving anaphoric expressions and fewer on resolving associative expressions and handling discourse-new expressions. The most frequently studied case of anaphoric reference is that of detecting and labeling co-reference relations, where one finds a set of linguistic expressions that denote the same referent. Anaphoric expressions, however, can also range over higher level linguistic constructions in discourse, such as discussed in Byron (2002) in the case on discourse deictic expressions and abstract anaphora [Byron, 2002]. Also definite discourse-new expressions can refer to contextually evocable entities, e.g. *the old bridge* or *the mountain* in Example (8b).

(8) (a) User: and then I'd like to get out of the - out of the downtown for a while and go on the, uh, philosopher's walk. Uh, how - how might I get there?
(b) WoZ: Um, you just, um, walk down the Haspelgasse in the opposite direction, and then you get to the old bridge. You just cross that, and then there're signs leading up to the mountain that's the philospher's walk.

Following Byron (2002) a discourse model contains so-called *discourse entities*. Discourse entities enter into the discourse model as information about events, objects, situations etc. is introduced into the discourse. Co-reference, then, means that a form part of an utterance, such as a pronoun, refers to a discourse entity that is already present in the discourse model. Setting up a referent in the discourse model, however, is not a trivial matter. In order to set up a referent correctly one has to solve various kinds of semantic and pragmatic problems that have been discussed in linguistic research falling under the categories of polysemy, metonymy, one-pronominalization, gapping and other forms of so-called *non-literal* expressions such as metaphoric expressions that will be discussed in more detail in Section 2.7.2.

However, as noted numerously in the literature natural language permits speakers to coerce terms in various ways. Coercion effects, in turn, affect pronominalization and, therefore, the resolution of anaphoric expressions. The consequence is that, unless, the discourse entities corresponding to the discourse-new expression are set-up correctly in the discourse model, anaphoric and other co-referential relations will become unresolvable by recourse to discourse context alone - for example in all cases where (grammatical) gender between the metonymic expression and the target referent differ.

Speakers can, therefore, employ extra-linguistic domain knowledge to introduce discourse-new discourse entities with definite articles, as in the case of metonymy or situational knowledge in the case of situationally-evoked referents. The same knowledge stores can be used to produce elisions and *contextual anaphora*, as in Example (9).

(9) a) User: Where is the castle?
b) WoZ: (spatial instructions)
c) User: How much does it cost?

As noted in linguistic analysis [Nunberg, 1987, Hobbs, 1991, Markert, 1999] metonymy and bridging phenomena are grounded on the fact that the given form and the referent exhibit a specific relation, called *pragmatic function* by Nunberg (1987), e.g. that museums feature exhibits licenses the bridge found in Example (7c). A bridging expression such as found in Example (10) bases on the same relation between tourist sites and their fees as the anaphora in Example (9c) exemplifies.

(10) The most popular site is the Heidelberg castle. The admission fee is 2 Euros.

For Bunt (2001) the relations between linguistic expressions and contextual settings - e.g. in the case of indexical expressions - are:

(a)expressions encoding or seeking information about aspects of contexts, e.g. about objects introduced earlier, situationally evoked referents or the relative time of speaking
(b)expressions that carry presuppositions, conversational implicatures and mappings based on shared beliefs and knowledge

In both cases the partial information encoded by the linguistic expression must be explicated relative to the given context in order for the expression to have a fully determined meaning. Expressions in which one finds (a) or (b) can thus only be understood through the relations between linguistic aspects and aspects of context. It follows that at least 90% of all declarative utterances cannot be understood if some information provided by contextual information and the corresponding knowledge is missing. In the following,I will show how the challenges have been addressed in the field of natural language processing.

2.4 Context in Natural Language Processing

Following [Allen et al., 2001b], one can differentiate between controlled and conversational dialogue systems. Since controlled and restricted interactions between the user and the system decrease recognition and understanding errors, such systems are reliable enough to be deployed in various real world applications, e.g.

timetable or cinema information systems. The more conversational a dialogue system becomes, the less predictable are the users' utterances. Recognition and processing become increasingly difficult and unreliable. This is due to the fact that on virtually all levels in the natural language processing pipeline, ambiguities, underspecification and noise multiply greatly.

Numerous research projects struggled to overcome the problems arising with more conversational dialogue systems [Allen et al., 2000, Malaka and Porzel, 2000, Johnston et al., 2002, Wahlster, 2003, Boves, 2004]. Their goals are more intuitive and conversational natural language interfaces that can someday be used in real world applications. The work described herein is part of that larger undertaking as I view the handling of contextual - and therefore linguistically implicit - information & knowledge as one of major challenges for understanding conversational utterances in complex dialogue systems. For this we will outline the various ways of dealing with context proposed in the literature and how context-dependent processing has been implemented in systems that seek to understand natural language input.

As in different fields of linguistics, e.g. pragmatics, cognitive-, socio- and psycholinguistics, the relations between utterances and context are also of concern to computational approaches. These have to specify how to compute the relations between linguistic and contextual aspects. This is important for both natural language understanding as well as generation. In understanding the question is how to *decode* the context-dependent aspects of a linguistic expression. In generation one wants to encode contextual information into the linguistic expression.

2.4.1 From Past to Present: The Historical Context

With its beginnings in the 1960s the first NLU systems drew primarily on lexical and syntactic recourses and aimed at recognizing patters that had specific significances for the target applications. Semantics in those systems was constituted by the application-specific significances of certain words and phrases or domain-specific categories as elements of *semantic grammars*, e.g. the PLANES [Waltz, 1978] or LIFER/LADDER [Hendrix, 1977] systems. First considerations of contexts emerged with the first attempts to build more realistic NLU systems starting with SHRDLU [Winograd, 1972] and LUNAR [Woods, 1977]. In these systems syntactic and semantic rules were used to parse utterances into components and to compute the ensuing consequences for the system. Only SHRDLU performed some dialogue functions and some context-dependent analysis restricted to discourse context. Experimental systems hence have increased their capabilities and involved contextual analysis as shown in Table 2.1.

Visible in all these experimental systems that were limited to such an impoverished contextual analysis and precompilations, was their restrictedness in terms of understanding capabilities, rendering them unscalable and in the case of more conversational input undeployable. This evidently shows up in the fragility of systems

Table 2.1 Employment of context in early dialog systems

Context	Usage
Domain Knowledge	lexicon building and syntactic categories
(static)	communicative function, e.g. speech acts
Discourse Knowledge	dialogue states and action planning
(dynamic)	reference resolution, e.g. anaphora

that fail when confronted with imperfect or unanticipated input, usually that also includes perfectly unambiguous utterance that stray but a little from a scripted demo dialogue. As noted above human conversations are between partners that share a rich background of contextual knowledge (some more static & some more dynamic contexts) without which natural language utterances become ambiguous, vague and informationally incomplete.

An interpreter with little context awareness and interpretation will encounter problems and fail frequently; one which does not fail in unexpected or problematic situations is called *robust*. Several means have been used to increase robustness as listed in Table 2.2. These so-called low-level techniques [Bunt, 2000] have not solved the problem of enabling system to react felicitously in a dynamic context. These techniques fail to assume a pragmatics-based approach where fact that the user has an intention communicated via a message where the intend has to be reconstructed by recourse to the current context. Advances in contextual analysis have been implemented in a handful of systems that increased their capabilities and involved contextual analysis insofar as domain ontologies have been employed for lexicon building, syntactic categories and semantic parsing and discourse knowledge for modeling dialogue states, action planning as well as for resolving anaphora and ellipsis.

Table 2.2 Means to increase robustness of early dialog systems

Object	Method
grammar	special rules and relaxations as well as automatic acquisition of semantic grammars
textual input	automatic spelling correction
lexica	on-line lexical acquisition

For example, the PHILQA system [Bronnenberger et al., 1997] featured context independent syntactic and semantic analyses as well as underspecified representations and context-dependent resolution with respect to the given domain representation. The SPICOS and TENDUM systems featured a resolution of structural ambiguity with underspecification, mass/count quantification with metavariables, and communicative functions determined by the user context [Bunt, 1984, Deemter et al., 1985]. Contextual underspecification was enabled by quasi-logical forms without semantic definition, which were instantiated unambiguously later by recourse to the semantic domain context, e.g. as implemented in the CLE system

[Alshawi and Moore, 1992]. In much the same way the influential TRAINS and TRIPS systems used unscoped logical forms as well as speech acts with context represented as user/system beliefs [Allen et al., 1995, Ferguson and Allen, 1998]. While these systems put a main focus on spatial domains helping users to solve specific tasks and produced considerable progress through developing corpora and NLP components the main emphasis rested on the planning part of the system.

Other systems employed dialog acts and thematic structures to decontextualize underspecified semantic representations or logical forms, such as VERBMOBIL [Wahlster et al., 1993] and PLUS/DENK [Bunt, 1989]. Given the distinction between global (unchanging or hardly changing) context, i.e. domain/world knowledge and local (changing) context, about the situation, user beliefs, system intentions or discourse, contextual considerations have either looked at utterances as a whole [Searle, 1975, Allen and Perrault, 1986, Perrault, 1989, Ramsey, 2000] or focused on reference & anaphora resolution [Grosz et al., 1977, Webber, 1991, Byron, 2002, Poesio, 2002]. On a rather general level particular computational linguistic knowledge sources can be organized into context-variant and -invariant ones as shown in Table 2.3 [Porzel and Strube, 2002].

Table 2.3 Context-variant and -invariant levels of analysis

	Context-variant	Context-invariant
Speech Recognition	vocabulary language model	basic vocabulary
Syntax & Parsing	open class lexicon parsing	closed class lexicon grammar
Semantics	disambiguation domain knowledge	lexical semantics á la DRT common-sense knowledge
Pragmatics	intention recognition	dialogue acts

2.4.2 The Present: Multimodal Systems

Enormous contributions to the field of Computational Linguistics come from attempts that focus on aiding human-human communication. Research systems such as Verbmobil and the C-STAR translators or various commercial translation systems [Hahn and Amtrup, 1996, Cettolo et al., 1999, Bub and Schwinn, 1999] have created architectures, standards and principles which also feature discourse context-sensitive understanding of an utterance's meaning [Pinkal et al., 2000]. However, that is not always the same as understanding the underlying intention, when the system has to *answer* to this input. There are several academic and commercial tools available which include information extraction systems, information retrieval systems, knowledge acquisition systems, spell-checker, auto summarizer or dictation systems. Usually these tools are seen as components of NLP systems and not as systems on their own. Previous systems focusing on human-computer interaction are by and large either focused on sophisticating their natural language input (understanding) side or their output (production) side. An additional common characteristic of existing systems is that they are bound to single, specific domains and their employment of (a priori) defined scripts for dialog management. However, several end-to-end spoken dialog systems and multimodal research prototypes exist. Most notably, the TRAINS system and its successor TRIPS [Ferguson and Allen, 1998] constitutes such a spoken dialogue systems which attempts to help users to solve tasks. Though this attempt involved a considerable amount of work in developing corpora and NLP components, the main emphasis lies on the planning part of the system [Allen et al., 1996]. Also, both systems deal with tiny domains. The AT&T telephone-based system *May I help you?* [Gorin et al., 1997] is – like the majority of spoken dialogue systems coming out of AT&T – restricted to a single domain with not much more than a dozen conversational topics. The same is also true for EVAR [Gallwitz et al., 1998] and the Philips train timetable system [Aust et al., 1995].

Multimodal dialogue systems as the QuickSet system [Cohen et al., 1997], the Command Talk spoken language system [Stent et al., 1999], the EMBASSI system [Herfet et al., 2001] or the Match system [Johnston et al., 2002] are quite narrow in focus and coverage of speech input. The vocabulary of these systems covers only a few hundred entries and the domain knowledge contains only a few dozen concepts. These systems allow interaction only in a very controlled fashion. In general the follow identical architectural pipelines, which have been generalized in the EMBASSI framework as shown in Figure 2.3.

This architecture consists of parallel input modalities (denoted by the letter I_n), which can be realized by automatic speech recognition systems, gesture recognition system or a graphical user interface. Their output is handed over to the respective modality-specific analyzers (denoted by the letter F_n). Correspondingly the modality-specific system responses are generated by a set of renderers (denoted by the letter R_n) and communicated via the modality-specific output mechanisms, such as graphical output or speech synthesis (denoted by the letter O_n). The task of multimodal fusion - unifying the input of the analyzers - and multimodal fission - distributing the output unto the renderers - is performed by the corresponding fusion and fission modules (denoted by *PMI* and *PMO* respectively). Ignoring the assistance and execution systems described in this architecture, the remaining part consists of a context manager, which obtains its input from biometric and sensoric input devices and stored information about the connected applications, devices in the environment and the user's preferences.

The SmartKom system [Wahlster et al., 2001] comprises a large set of input and output modalities which the most advanced current systems feature, together with an efficient fusion and fission pipeline. SmartKom features speech input with prosodic analysis, gesture input via infrared camera, recognition of facial expressions and their emotional states. On the output side, the system features a gesturing and speaking life-like character together with displayed generated text and multimedia graph-

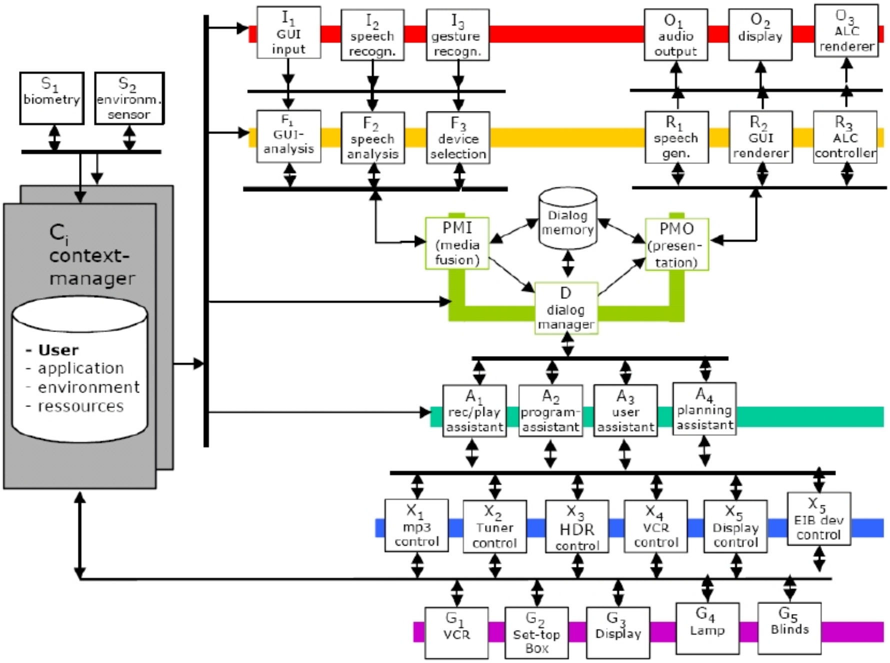

Fig. 2.3 The EMBASSI multimodal architecture

ical output. It comprises nearly 50 modules running on a parallel virtual machine-based integration software called *Multiplatform*[3] and shown in Figure 2.4.

2.5 Methodological Background

In this section I will present methodological approaches for evaluating the performance dialog-, speech- and discourse understanding systems in the light of their pertinence for the evaluations performed in this work as well as their respective state of the art. Therefore, I will sketch out the most frequently used metrics for evaluating the performances of the relevant components and systems at hand in terms of their pertinence and applicability, focusing also on the specific contribution to this field that were brought about as a result of the measurements and metrics adopted in this work.

[3] The abbreviation stands for "MUltiple Language / Target Integration PLATform FOR Modules".

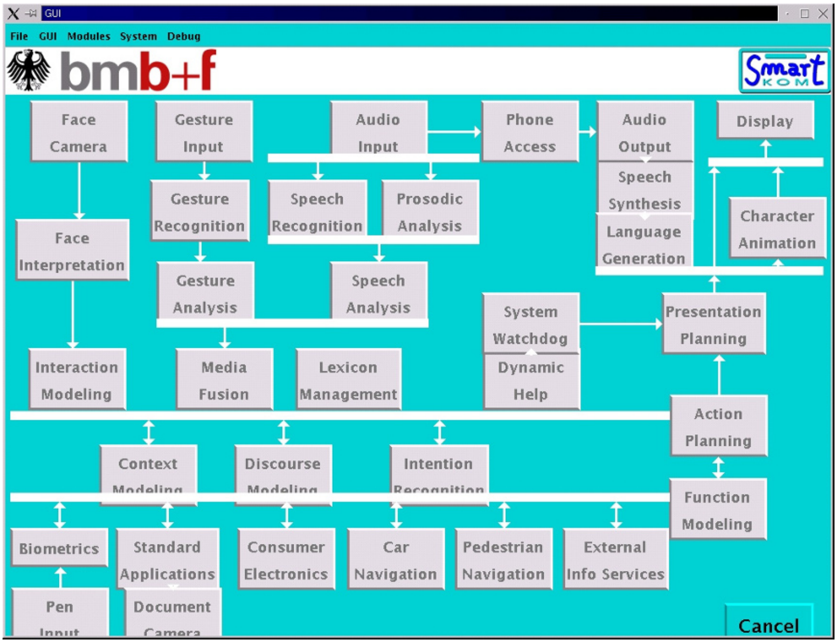

Fig. 2.4 The SmartKom multimodal architecture

2.5.1 Performance in Dialogue Systems Evaluations

For evaluation of the overall performance of a dialogue system as a whole frameworks such as PARADISE [Walker et al., 2000] for unimodal and PROMISE for multimodal systems [Beringer et al., 2002] have set a *de facto* standard. These frameworks differentiate between:

- dialogue efficiency metrics, i.e. elapsed time, system- and user turns
- dialogue quality metrics, mean recognition score and absolute number as well as percentages of timeouts, rejections, helps, cancels, and barge-ins,
- task success metrics, task completion (per survey)
- user satisfaction metrics (per survey)

These metrics are crucial for evaluating the aggregate performance of the individual components, they cannot, however, determine the amount of understanding *versus* misunderstanding or the system-specific *a priori* difficulty of the understanding task. Their importance, however, will remain undiminished, as ways of determining such global parameters are vital to determining the aggregate usefulness and felicity of a system as a whole. At the same time individual components and ensembles thereof - such as the performance of the uni- or multimodal input understanding system - need to be evaluated as well to determine bottlenecks and weak links in the discourse understanding processing chain.

2.5.2 Performance in Automatic Speech Recognition Evaluations

The commonly used word error rate (WER) can be calculated by aligning any two sets word sequences and adding the number of substitutions S, deletions D and insertions I. The WER is then given by the following formula where N is the total number of words in the test set.

$$WER = \frac{S+D+I}{N} \times 100 \tag{2.1}$$

Another measure of accuracy that is frequently used is the so called *Out Of Vocabulary* (OOV) measure, which represents the percentage of words that was not recognized despite their lexical coverage. WER and OOV are commonly intertwined together with the combined acoustic- and language-model confidence scores, which are constituted by the posterior probabilities of the hidden Markov chains and n-gram frequencies. Together these scores enable evaluators to measure the absolute performance of a given speech recognition system. In order to arrive at a measure that is relative to the given task-difficulty, this difficulty must also be calculated, which can be done by means of measuring the perplexity of the task see Section 2.6.

2.5.3 Performance in Understanding Evaluations

A measure for understanding rates - called *concept error rate* has been proposed for example by Chotimongcol and Rudnicky (2001) and is designed in analogy to word error rates employed in automatic speech recognition that are combined with keyword spotting systems [Chotimongcol and Rudnicky, 2001]. They propose to differentiate whether the erroneous *concept* occurs in a *non-concept slot* that contains information that is captured in the grammar but not considered relevant for selecting a system action (e.g., politeness markers, such as *please*), in a *value-insensitive slot* whose identity, suffices to produce a system action (e.g., affirmatives such as *yes*), or in a *value-sensitive slot* for which both the occurrence and the value of the slot are important (e.g., a goal object, such as *Heidelberg*). An alternative proposal for concept error rates is embedded into the speech recognition and intention spotting system by Lumenvox[4], wherein two types of errors and two types of non-errors for concept *transcriptions* are proposed:

- A *match* when the application returned the correct concept and an *out of grammar match* when the application returned no concepts, or discarded the returned concepts because the user failed to say any concept covered by the grammar.
- A *grammar mismatch* when the application returned the incorrect concept, but the user said a concept covered by the grammar and an *out of grammar mismatch* when the application returned a concept, and chose that concept as a correct interpretation, but the user did not say a concept covered by the grammar.

[4] www.lomunevox.com/support/tunerhelp/Tuning/Concept _Transcription.htm

Neither of these measures are suitable for our purposes as they are known to be feasible only for context-insensitive applications that do not include discourse models, implicit domain-specific information and other contextual knowledge as discussed in Porzel et al [Porzel et al., 2006a]. Therefore this measure has also been called *keyword recognition rate* for single utterance systems. Another crucial shortcoming noted [Porzel and Malaka, 2004b], is the lack of comparability, as these measures do not take the general difficulty of the understanding tasks into account. Again, this has been realized in the automatic speech recognition community and led to the so called *perplexity* measurements for a given speech recognition task. I will, therefore, sketch out the commonly employed perplexity measurements in Section 2.6.

The most detailed evaluation scheme for discourse comprehension, introduced by Higashinaka *et al* (2002), features the metrics displayed in Table 2.4. Higashinaka *et al* (2003) combined these metrics by means of composing a weighted sum of the results of multiple linear regression and a support-vector regression approaches [Higashinaka et al., 2003]. This sum is. then, compared to human intuition judgments and metrics, comparable to PARADISE metrics [Walker et al., 2000], concerning task completion rates and -times. While this promising approach manages to combine factors related to speech recognition, interpretation and discourse modeling, there are some shortcomings that stem from the fact that this schema was developed for single-domain systems that employ frame-based attribute value pairs for representing the user's intent.

Table 2.4 Proposed measurements of discourse comprehension [Higashinaka et al., 2002]

1	slot accuracy
2	insertion error rate
3	deletion error rate
4	substitution error rate
5	slot error rate
6	update precision
7	update insertion error rate
8	update deletion error rate
9	update substitution error rate
10	speech understanding rate
11	slot accuracy for filled slots
12	deletion error rate for filled slots
13	substitution error rate for filled slots

Nevertheless, recent advances in discourse modeling, as described in Section 3.1.2 together with multi-domain systems enable approaches that are more flexible and more difficult to evaluate than the slot-filling measures described above, as they employ discourse pegs, dialogue games and overlay operations [Pfleger et al., 2002, Alexandersson and Becker, 2003] for handling more conversational input and cross-modal references . More importantly, no means of measuring the *a priori* discourse understanding difficulty is given, as I will discuss in Section 2.6.

2.5.4 Performance in Classification Evaluations

In the realm of semantic analyses the task of word sense disambiguation is usually regarded as the most difficult one. This means it can only be solved after all other problems involved in language understanding have been resolved as well. The hierarchical nature and interdependencies of the various tasks are mirrored in the results of the corresponding competitive evaluation tracts - e.g. the message understanding conference (MUC) or SENSEVAL competition. It becomes obvious that the ungraceful degradation of f-measure scores (shown in Table 2.5.4 is due to the fact that each higher-level task inherits the imprecisions and omissions of the previous ones, e.g. errors in the named entity recognition (NE) task cause recall and precision declines in the template element task (TE), which, in turn, thwart successful template relation task performance (TR) as well as the most difficult scenario template (ST) and co-reference task (CO). This decline can be seen in Table 2.5.4 that presents their corresponding f-measures - where precision and recall are weighted equally as given by the Formula 2.2 below [Marsh and Perzanowski, 1999].

Table 2.5 Evaluation results of the best systems of the 7th Message Understanding Conference

NE	CO	TE	TR	ST
$f \leq .94$	$f \leq .62$	$f \leq .87$	$f \leq .76$	$f \leq .51$

Despite several problems stemming from the prerequisite to craft costly gold standards, e.g. tree banks or annotated test corpora, precision and recall and their weighable combinations in the corresponding f-measures (such as given in Table 2.5.4), have become a *de facto* standard for measuring the performance of classification and retrieval tasks [Van Rijsbergen, 1979]. Precision *p* states the percentage of correctly tagged (or classified) entities of all tagged/classified entities, whereas recall *r* states the positive percentage of entities tagged/classified as compared to the normative amount, i.e. those that ought to have been tagged or classified. Together these are combinable to an overall f-measure score, defined as:

$$F = \frac{1}{\alpha\frac{1}{p} + (1-\alpha)\frac{1}{r}} \tag{2.2}$$

Herein α can be set to reflect the respective importance of *p versus r*, if $\alpha = 0.5$ then both are weighted equally. These measures are commonly employed for evaluating part-of-speech tagging, shallow parsing, reference resolution tasks and information retrieval tasks and sub-tasks.

An additional problem with this method is that most natural language understanding systems that perform deeper semantic analyses produce representations often based on individual grammar formalisms and mark-up languages for which no gold standards exist. For evaluating discourse understanding systems, however, such gold standards and annotated training corpora will continue to be needed.

2.6 Measuring Task Difficulties and Baselines

As the measurements, presented in Section 2.5.3, are not designed to reflect complexity of the tasks performed by the relevant components and systems at hand. I will, therefore, present the most frequently used metrics for estimating the difficulty inherent in such tasks as will be pertinent herein.

2.6.1 Measuring Perplexity in Automatic Speech Recognition

Perplexity is a measure of the probability weighted average number of words that may follow after a given word [Hirschman and Thompson, 1997]. In order to calculate the perplexity B, the word entropy H needs to be given for the specific language of the system W. The perplexity is then defined within limits as:

$$0 < H = -\sum_{\forall 1 < W < n} P(W) log_2 P(W) < log_2 n \qquad (2.3)$$

$$B = 2^H$$

Improvements of specific speech recognition systems can then consequently be measured on a corpus with a given perplexity by measuring the corresponding error rates (WER and OOV) Together, this yields a performance measure for recognition quality that can be compared to other speech recognition performances on corpora with differing perplexity. The more common approach is to employ baseline measurements as a comparison for individual performances, e.g., where perplexity measures or other task-difficulty metrics are not at hand, as it is usually the case in classification tasks. I will, consequently, present pertinent baseline approaches in the following section.

2.6.2 Measuring Task-specific Baselines

Baselines for the performance of classification tasks are commonly defined based on chance performance, on an *a posteriori* computed majority class performance or against the performance of an established classification method. In other words, using the f-measure for performance discussed in Section 2.5.4, one can ask:

- what is the corresponding f-measure, if the evaluated component guesses randomly - for chance performance metrics,
- what is the corresponding f-measure if the evaluated component always chooses the most frequent solution - for majority class performance metrics,
- what is the corresponding f-measure of the established baseline classification method.

Much like kappa coefficiency measures for statistical inter-rater agreement, where observed agreement $P(a)$ is set in relation to what one would have expected $P(e)$ as shown in Formula 2.4 [Galton, 1892, Cohen, 1960, Carletta, 1996], existing employments of majority class baselines assume an equal set of identical potential mark-ups, i.e. attributes and their values, for all markables.

$$\kappa = \frac{P(a) - P(e)}{1 - P(e)} \tag{2.4}$$

Therefore, they cannot be used in a straight forward manner for many tasks that involve disjunct sets of attributes and values in terms of the type and number of attributes and their values involved in the classification task. This, however, is exactly what we find in natural language understanding tasks, such as in so-called *sense tagging* or *word sense disambiguation* tasks [Stevenson, 2003]. Additionally, baseline computed on other methods cannot serve as a means for measuring scalability, because of the circularity involved: as one would need a way of measuring the baseline method's scalability factor in the first place. Table 2.6.2 provides an overview of the existing ways of measuring performance and task difficulty in automatic speech recognition and understanding.

Table 2.6 Summary of performance and difficulty measurements

Domain	Performance	Difficulty
automatic speech recognition	WER/OVV	Perplexity
natural language understanding	CER	none
MUC tasks (NE, TE, TR, ST, CO)	f-measure	baselines
unimodal dialogue system	PARADISE	none
multimodal dialogue system	PARADISE	none

Current evaluation frameworks for uni- or multimodal dialogue systems that allow for spoken language input do not include metrices for measuring the quality of the intention recognition [Walker et al., 2000, Beringer et al., 2002], simply because such information is hard to extract automatically from the logs of system runs [Litman et al., 1999b]. Furthermore, no general computational method or framework for measuring the difficulty of natural language understanding tasks have been proposed so far. We are, therefore, faced with a lack of methods for measuring the difficulties of the individual tasks involved in the language understanding process.

Such generally applicable methods, however, are needed for measuring the scalability of natural language understanding systems and components.

2.7 Point of Departure

Utterances in dialogues, whether in human-human interaction or human-computer interaction, occur in a specific situation that is composed of different types of contexts. In the following a categorization of the types of context relevant to spoken dialogue systems - and human computer interaction in general - is given together with their respective scope (content) and modularization in HCI systems. This work will, as observable below, depart from the common distinction between linguistic and extra-linguistic contexts, whereby all extra-linguistic contexts are also often lumped together as the *situational context* [Connolly, 2001]. The categorization proposed herein subsumes the linguistic context under the heading of *dialogical context*. Dialogical context encompasses the dialogical counterparts of both co-text and inter-text as well as non-linguistic input from other modalities, e.g. interacting with traditional interfaces (WIMP) or gesture and the like. The categorization employed herein also differentiates extra-linguistic (or situational) context into interlocutionary-, domain- and situational context as shown below.

2.7.1 Context Types

As shown in table 2.7 dialogical context in our model corresponds to what has been termed *linguistic context* in the domain of natural language analysis and encompasses information from the discourse history, i.e. prior utterances by the interlocutors. As pointed out in section 2 discourse context is essential for a variety of tasks that one finds under the headings of *reference resolution*, *anaphora resolution* or *semantic disambiguation*. The following section presents an elaboration of these problems in the light of the essential contributions of context.

Table 2.7 Contexts, content and knowledge sources

types of context	content	knowledge store
domain context	world/conceptual knowledge	domain model
dialogical context	what has been done by whom	dialogue model
interlocutionary context	properties of the interlocutors	user model
situational context	time, place, etc	situation model

2.7.2 The Tasks (Revisited)

In order to employ a consistent terminology in the subsequent discussions and experiments on finding appropriate meanings for given linguistics forms, I will adopt - wherever possible- the basic notations and insights that originated in the so called *construction grammar* framework [Lakoff, 1987, Langacker, 1987, Fillmore, 1988, Talmy, 1988]. As also noted in Section 4 work on context and real language use in formal linguistics was based the earlier insights in functional and usage-based models of language and was mainly restricted to the field of Cognitive Linguists.

The ensuing grammatical framework and vocabulary in formal construction grammar [Goldberg, 1995, Kay and Fillmore, 1999, Feldman, 2006], has been explicitly devised to handle actually occurring natural language phenomena, which notoriously contains non-literal, elliptic, context-dependent, metaphorical or underspecified linguistic expressions. As shown in this chapter, these phenomena still present a real challenge for current natural language understanding systems. Furthermore, I agree with the central principle of construction grammar which states that grammatical phenomena also contribute to the meaning of a sentence which is the reason why syntax cannot be defined independently of semantics of a grammar.

Constructions are the basic building blocks, posited by the construction grammar framework, and are defined as follows [Goldberg, 1995]: A construction is a form-meaning pair $< F_i, S_i >$ if some aspect of F_i or some aspect of S_i is not strictly predictable from the component parts of that construction or from other previously established constructions. (Ibid:4). Using this framework, the aforementioned task of resolving referring expressions and ambiguities can be stated as follows: Given a form F_i, which can be a referring expression such as *the bank* or an anaphora such *it*, what is the corresponding meaning $S(F)$ in the given context. It is important to keep in mind that a form in construction grammar can be constituted on all linguistic levels, i.e. we find phonological forms, morpho-syntactic forms, lexical forms and clausal forms. That means, one can describe lexical constructions as *the* and *bank* individually, look at a composite construction such as the referring expression *the bank* or even a whole utterance such as *Where is the bank* and how they can *resolved*, meaning which specific meaning is to be assigned to it given the context at hand.

Computationally, this entails - as I will show in Sections 3.2 through 3.5 as well as in Section 4.2 - dealing with our target challenges in automatic language understanding for resolving a contextually adequate formal specification of the semantics from the given ensemble of forms. The target data structure will, consequently, be referred to as a *semantic specification* [Chang et al., 2002]. The corresponding tasks in natural language generation are selecting (constructing) in a context-dependent manner - a semantic specification out of myriads of alternatives and the ensuing construction-based formulation thereof.

Numerous works have sought to label various relations between forms and meanings. I have already exemplified the difference between homonymy and polysemy, but additional phenomena have received great attention, e.g. metonymy [Hobbs, 1991, Markert, 1999], metaphor [Lakoff and Johnson, 1980], coercion and type shifting [Faucounnier and Turner, 1998, Michaelis, 2001] as well as mental spaces [Fauconnier, 1985]. While this work will not discuss these phenomenon in greater detail, it is important to note that the fundamental assumption underlying such analysis is that individual forms feature some kind of literal meaning and that they can assume non-literal meaning by means of metonymical or metaphorical usage, coercion and the like. While the work presented herein departs from this assumption, our view is not irreconcilable with it.

Various terms have been proposed in the literature, e.g., *intricacy* or entrenchment [Faucounnier and Turner, 1998, Langacker, 2000], that express that these phenomena can be measured on a scale. Which means, in the words of Faucounnier and Turner, that meaning can be assigned to forms with increasing or decreasing intricacy. A *literal* usage would require little to no intricacy and others, such as so-called *blends* require more (ibid). For Langacker this intricacy can be boiled down to statistical measure of *entrenchment*, the more frequently used a specific contextually evoked form - meaning pairing becomes, the more central is the correspondingly entrenched meaning of that form [Langacker, 2000]. Before returning to these questions below, let me point out, once more, that the general task of determining a particular meaning representation - that has to be constructed with more or less intricacy from the forms at hand - will be the central empirical domain to be employed in the approach to contextual computing presented herein. If linguistic forms were to be unambiguous and always fully specific, then no additional meaning construction would be necessary due to the given one-to-one mapping between a specific form and its meaning. Since that is, obviously, not the case additional information and knowledge is needed to construe the intended meaning of a given form.

In the computational sense this entails that meaning resolution can be seen as determining the most plausible meaning from the set of the possible meanings that can be constructed out of the given form.[5] In line with the central claim of this work, we, therefore, find that on all computational levels of natural language processing where underspecification, ambiguity and noise arises, one needs additional information and knowledge for finding the most plausibly constructed meaning, thereby resolving the form-meaning mappings out of many other potential ones. As I will argue below contextual information and pragmatic knowledge constitutes this additional modality by means of which meaning resolution becomes possible, or - in other words - the other potential form-meaning mappings are inhibited from being activated as the most plausible one did.

What would, therefore, be needed is a context-dependent scoring that identifies the most plausible item out of a set of possible alternatives. In the following sections I will introduce, examine and evaluate how such an approach to contextual computing can be employed to increase performance, robustness and scalability of natural language processing systems in the areas of:

- Automatic Speech Recognition
- Semantic Interpretation
- Pragmatic Interpretation

[5] This, in turn, is quite congruent to other approaches in contextual computing where the computational notion of *correctness* is - by necessity - replaced with the notion of *plausibility*.

Before I present the data, experiments and results of applying the approach to contextual computing pursued herein to these areas of natural language processing, I will discuss the subsequent modeling of contextual knowledge stores employed by the contextual computing approach to be discussed hereafter. Since domain knowledge is nowadays commonly modeled using formal ontologies, they will be introduced first generally and then specifically, in terms of the concrete ontologies and modeling principles employed to represent domain knowledge in our experiments.

http://www.springer.com/978-3-642-17395-0

Contextual Computing
Models and Applications
Porzel, R.
2011, XI, 180 p., Hardcover
ISBN: 978-3-642-17395-0