Invited Paper

Language, Representation And Contexts

R. V. GUHA* and D. B. LENAT*

In this article, we will examine the inherent infeasibility of the traditional model of AI, and propose some modifications. We will examine some of the fundamental assumptions made by this model and the problems they lead to . We will then introduce the concept of *Contexts* as a solution to these problems and describe a context-based framework for knowledge representation and natural language understanding.

1. The Symbolic Model of AI

Most architectures that have been proposed for symbolic AI systems [5, 10] share the following theme. There is a repository of knowledge, the Knowledge Base (KB), and a set of procedures (the problem solver or inference engine), which operate on it to produce intelligent behavior. Inputs to the system are translated by an appropriate front end into the language of the KB before being added to it.

In the currently dominant school of thought, this KB uses a declarative encoding (an encoding whose meaning is independent of the programs that operate on it) and the procedures carry out deductions. The system has some domain of competence; the overall goal of AI is to build a system whose domain of competence is comparable to that of humans (at least in some domains). The KB primarily contains the system's knowledge about its domain of competence. Occasionally the KB may also have some meta-knowledge about how to use this knowledge [6].

In this article, we will examine the inherent infeasibility of this model such as it is, and propose some modifications. We will examine some of the fundamental assumptions made by this model and the problems they lead to. We will then introduce the concept of *Contexts* as a solution to these problems and describe a context-based framework for knowledge representation and natural language understanding.

2. Critique of the Traditional Model

The KB consists of a set of expressions (sentences) in a certain vocabulary. Each sentence conveys some truth about the domain. In the ideal case, the meaningfulness and truth of each sentence is independent of the presence or absence of other sentences. To use Quine's terminology [13, 14], these are assumed to be *eternal* or *universal* sentences, i.e., all foreseeable dependencies (of the sentence) have been made explicit in the sentence. E.g., the sentence "There are four billion people alive on Earth" is not eternal - it assumes that the time is 1991.[1] Making a sentence *eternal* requires completely *decontextualizing* it, i.e., making all foreseeable dependencies (of the sentence) explicit in the sentence.

Sentences in such a KB are thus very different from natural language (NL) utterances. NL utterances are far from being *universal.* They usually make a number of assumptions and depend very heavily on situational factors to convey their intended meaning. These situational factors could potentially include not just previous utterances, but also broader factors such as the goals of the discoursants, the socio-cultural setting of the discourse, etc.

2.1 Problems with Representation in the Traditional Model

One of the first things we need to do in representing any domain is to pick a vocabulary for encoding the KB's knowledge of that domain. The vocabulary should allow expression of facts about the phenomena we expect to find. The choice of the vocabulary can therefore exclude certain phenomena from consideration by the KB. If certain phenomena are excluded, this constitutes an assumption that these phenomena are not important/do not exist. The exclusion of these could either be by design or simply accidental. It is almost inevitable that certain parts of the domain will be overlooked in the representation process.

Consider representing a theory of commercial transactions. More specifically, assume we are interested in representing the concepts of buying and selling. We decide that to refer to the event of X buying Y from Z,

*CYC PROJECT, MCC, 3500 W. Balcones Center Dr., Austin, Texas, 78759.

[1]Even making the time explicit to the nearest nanosecond does not make this sentence fully decontextualized. It makes assumptions about who is considered a 'live person'—Are fetuses in the womb alive? Does it include people in orbit around the earth? . . .

we use the term (Buy X Y Z). We realize that this is insufficient since it cannot distinguish between two separate events involving X buying Y from Z (i.e., if Z sells Y to X, Z then buys it back from X and sells it to him again, both the first and third transaction will be (Buy X Y Z) so that we cannot distinguish between them.)

To uniquely identify the sale, we add an extra argument to represent the time at which the sale took place. So, to refer to X buying Y from Z at time T1, we use the term (Buy X Y Z T1). At first glance, this seems adequate in that it allows us to accurately refer to any particular buying event. However, even this excludes certain phenomena. Consider the following admittedly farfetched situation. Z has engaged two agents (computer programs) to sell Y. Both of these simultaneously make deals with X's agents (again, computer programs) to sell Y. Here we have two sales of Y to X at the same time, and our vocabulary does not allow us to distinguish between them.

We decide that the way out of this problem is to reify the action. Rather than writing (Buy X Y Z), we write Buying (Buying001)∧buyer (Buying001 X) ∧object (Buying001 Y)∧seller (Buying001 Z). If two simultaneous deals are made between Z and X to buy Y, these will be different buying actions (Buying001 and Buying002). If the price negotiated for Buying001 is P, we write price(Buying001) = P. Notice that though this new scheme based on the reification of events is more expressive, it is also significantly more cumbersome.

Unfortunately, it is still not the case that we have taken all possible (buying related) phenomena into account. Remember that we write price(Buying001) = P to state that the price paid was P. Now, someone may point out that this is ambiguous. In certain purchases, there might actually be many different prices involved (and it might not be possible to agglomerate them into a single term). For example, in addition to a sum of money, the buyer might also have a certain obligation towards the seller (e.g., in the case of one country selling another weapons) and it is not really possible to lump the obligation and money exchanged into a single term. By using a single concept of price, our vocabulary precludes us from considering these phenomena. So it is still the case that not all buying related phenomena are expressible.

We could of course further refine our vocabulary to cover these phenomena. But in the process we also make the vocabulary more awkward to use. This problem has certain similarities to the *Qualification Problem* [12]. In the Qualification Problem, the need to take more possibilities into account leads to more and more antecendents for rules. Similarly, the need to take more phenomena into account leads to increasingly cumbersome vocabularies (e.g., an increasing number of arguments to predicates and functions). Requiring a tradeoff between the extent of the domain that can be covered and the usability of the vocabulary seems undesirable.

It also seems that with a little searching, we can always uncover inexpressible phenomena. At some point, we have to finalize the vocabulary and carry on with the representation.

The obvious objection to this argument is that after a certain point, the excluded phenomena will be increasingly strange and unlikely to be encountered, so that it is therefore acceptable to ignore them. For example, the phenomenon of multiple simultaneous deals between the seller and buyer is indeed a little out of the ordinary. But, despite its strangeness, we (humans) can quite easily conceptualize this phenomenon. We might be somewhat at a loss to predict what exactly will happen, but we can certainly imagine and discuss it. Our programs should have the "upward compatibility" in their vocabularies that we have.

As with the vocabulary, the theory that we write to describe the domain also makes assumptions. In our theory of commercial transactions, we might have axioms which state that in a purchase, the buyer usually needs the object bought, the seller needs the money, after the sale the buyer owns the object, etc. Though these rules are very basic, they still make many assumptions. They assume a certain amount of rationality on the part of the buyer and seller. If the seller was engaging in the event for spite (e.g., a dying man selling his house for a dollar so that his children didn't get it), or if the buyer were engaging in the action as an act of charity, many of the axioms in the theory would be inapplicable. Any representation makes sense only in the context of a background theory, one which specifies the assumptions made by the theory. Attempts to completely decontextualize the theory, i.e., to incorporate all of the background theory into the domain theory, would be futile.

So, what is wrong if the domain theory makes a number of assumptions? The problem is that if we attempt to reduce significantly the dependence of the theory on its implicit assumptions, it gets too cumbersome to be really usable.

Both of these, (limitations of the vocabulary, and accounting for assumptions made by the theory), are problems we face in building a knowledge base. The standard advice is to develop and fix an adequately rich vocabulary that can express all the phenomena the application program is expected to encounter, and to ferret out and make explicit any assumption our theory might make. While this is good advice, it is not adequate as the only solution to these problems. Let us consider some of its shortcomings.

- It is almost inevitable that there will be some chafing limitations in the vocabulary and some limiting assumptions behind the theory. When these limitations are eventually discovered, extending the KB to deal with these phenomena could require a reworking of large parts of the KB. This is undesirable in that we would like (a) a more graceful way of extending the KB to deal with hitherto unexpected phenomena (b) not to lose the efficiency associated with simpler representation when dealing with simple cases, just to be able to handle a few more cases.

- A vocabulary (and theory) that attempts to have a very broad scope is very likely to be cumbersome. For example, in the case of buying, (Buying X Y Z) is much more compact than Buying (Buying001)∧buyer (Buying001 X)∧object (Buying001 Y)∧seller (Buying001 Z). This is undesirable both from the perspective of writing the axioms and from the perspective of efficient inference.

Even a naive theory of buying and selling that could not deal with simultaneous transactions and which made assumptions about the participants being rational would certainly be very useful. In fact, this theory is all we need for most purposes. At the same time, we humans can understand and cope with the more abnormal situations and the KB should similarly be capable of understanding these.

2.2 Problems from an NL Perspective

2.2.1 The Traditional Model of an NL Interface

Part of what it means to be intelligent is to be able to communicate in natural language (NL). The model described at the beginning posits a "front end" which takes natural language utterances and produces sentences for inclusion in the knowledge base. In the traditional model, these sentences must of course satisfy the requirement imposed on all the sentences in the KB, i.e., they must be completely decontextualized.

The operation of the front end (the NL program) is guided mostly by linguistic information. In cases where purely linguistic information is inadequate to disambiguate the meaning of an utterance, the NL program might call upon the knowledge base. This use of the knowledge base (in disambiguation) usually takes the form of the NL program generating the possible set of interpretations of an utterance and asking the knowledge base either to eliminate some or to rank them in order of relative likelihood.

The NL program might keep some state to help it resolve anaphora, ellipsis and other reference problems. This state usually consists of utterances previously made and the objects to which these utterances refered. It is normally expected that this state is adequate to resolve the context dependency of natural language utterances. At the end of the process, the NL program is expected to have *the* completely decontextualized translation which is then added to the KB.

2.2.2 Problem: Context Dependence in NL

There is an (implicit) assumption made that the context dependency of natural language utterances can be accounted for completely using linguistic knowledge. It is also assumed that the utterance can be completely decontextualized and that all the context dependencies of an utterance are revealed by the utterances preceding it.

Unfortunately, many of the contextual dependencies in NL utterances are coarser-grained than and unrelated to the contextual phenomena (such as anaphora, indexicals, etc.) that are studied by linguists. The following two examples illustrate the ubquity of coarse-grained context dependence.

(a) Consider the literature that accompanies a model airplane kit. Some of the material on the cover of the kit assumes the person has not yet decided to buy the product. Some of the instructions inside the box assume the person has bought the model and would like to assemble it. They also assume the person has no higher priority task (than assembling the kit) at the moment and would like the assembly to be capable of all the kit was advertised to be. There might be other instructions accompanying the model which specify what should be done in case of certain emergencies (such as a child swallowing the adhesive provided with the model). These instructions make assumptions that are radically different from those made by the assembly instructions. For example, the emergency instructions assume that the reader cares more about saving the child's life than about protecting or assembling the model.

(b) Consider a section of a travel advisory telling people what to expect, how to dress, etc. in the northeast of the United States during winter. Somewhere at the beginning, there might be the phrase 'during winter'. This sets up the context for the remainder of the advisory. Each following statement is assumed to inherit the temporal qualification 'during winter'. Similarly, statements about dress, temperature, etc. also implicitly concern the outdoors.

    There might be other similar but unstated temporal qualifications. For example, there might be a statement such as, "the snowplows run between 3 and 4 in the morning (in Boston)". This statement (like much of the rest of the section) assumes the implicit temporal qualification that it holds true around the time the advisory was published.

These examples of coarse grained contextual effects in natural language utterances are clearly outside the scope of linguistic theories and require domain knowledge for adequate processing.

The problem is that the system might not know *a priori* all the contextual aspects when approaching a text or a conversation, but instead may have to *infer* these as the discourse proceeds. The contextual aspects could be deduced from a *combination* of a very wide array of factors. They could be determined based on the utterances in the discourse, the actions of the speaker (and listener), the culture of the discourse participants, the place the discourse is occurring, etc.

What we have here is a "chicken and egg" situation. The system needs to use the information in the knowledge base together with a new assertion (the one just made by the speaker) to *infer* the context dependence of that new assertion. To effectively combine the information in the knowledge base with the new assertion, the assertion must already be in the knowledge base. But the system needs to factor out the context dependence of the assertion before it can be added into the knowledge base! The way out of this cycle is to introduce a mechanism that will allow us to add the assertion to the knowledge base without factoring out all the contextual effects.

2.2.3 Side Effects of NL based Pre-Formalization

For most domains (especially those not closely linked to perception), writing a theory (of the domain) in terms of formulas is usually preceded (and accompanied) by writing and discussing the theory in English. As a result, some aspects of natural language often remain in the resulting representation.

Example:

Two theories might each involve the concept of "seller." The English word "seller" is highly polysemous, i.e., has many very closely related but distinct meanings: the person with whom a buyer interacts and might ask questions, the shop the where the buying takes place, the organization owning the shop where the buying takes place, etc.

The concept "seller" may be represented in the knowledge base using a single predicate—*seller*-with the polysemous nature of the English word "seller" reflected by different theories using the predicate with these different English meanings as the intended denotation. The situation is especially bad with words that assume a very wide range (or continuum) of meanings. E.g., The term "USA" could be used to refer to any of the following: the United States government, the executive branch of the United States government, the judiciary branch of the United States government, the economy of the United States, the geographical region encompassing the United States, the defense forces of the United States, a delegation from the United States, . . .

There are many other side effects of a linguistic preformalization step in the representation process that we will not go into here. In general, these have the effect of the representation having implicit contextual dependencies similar to those in the pre-formalization NL utterances.

3. Contexts as a Solution to these Problems

We have described a diverse range of problems with the traditional model of AI. Given the fairly broad range of phenomena covered by these problems, the natural question that arises is whether these are very different problems requiring very different solutions.

Though these are certainly very different kinds of problems, they share a common origin, the intended universal and decontextualized nature of expressions used in representation. This suggests that at least at a logical level, a single machinery, one that allows for this requirement on representations to be relaxed, might be shared by the solutions to these different problems.

The general problem is that a representation (a set of axioms written as part of a theory of some topic or a set of axioms obtained as a translation of NL utterances) by itself does not contain all the information associated with it. The information associated with a representation of commercial transactions would have to include not just the axioms describing buying and selling, but also information about the assumptions made by the theory, about when these assumptions are reasonable, about when this theory is applicable, etc. These are "external" to the axioms constituting the theory itself.

If, in the statement of the theory (whether it be a set of axioms written by humans or the translation generated by an NL front end) we can make it clear that there is something (possibly as yet unstatable) left out, this theory could be included in the KB without ruling out the possibility of later extending the KB. We now consider a syntax for this.

Consider the axiom, "After the sale, the buyer owns the object". For convenience let us refer to this as A1. Let us name this theory the "NaiveMoneyMt".[1] To say that A1 is an axiom of NaiveMoneyMt, we write,

ist(NaiveMoneyMt A1)

ist stands for 'is true in'. The first argument to *ist* denotes a *Context*.

Similarly, given an utterance U1 in a text, if the literal translation (i.e., the translation provided by the NL front end into the system's logical representation) of this sentence is F1, the sentence

ist(UC(U1)F1)

is added to the KB, where UC(U1) is the context of the utterance U1.

3.1 What is a Context?

Contexts are objects in the ontology; i.e., we can make statements "about" contexts. They are rich objects[2] in that a context cannot be completely described. The contextual effects on an expression are often so rich that they cannot be captured completely in the logic.

[1]The suffix 'Mt' comes from the term *Microtheory*, used to refer to a special class of contexts in the Cyc system.

[2]A rich object cannot be defined or completely described. The system may be given facts about a rich object but never the complete description.

This is what leads us to incorporate contexts as objects in our ontology.

In the formula ist(NaiveMoneyMt A1), the context denoted by the symbol NaiveMoneyMt is supposed to capture everything that is not in A1 that is required to make A1 a meaningful statement representing what it is intended to state. This includes assumptions made by A1 about the domain, assumptions about the applicability of A1, how A1 might relate to statements in other contexts, etc.

The idea is that A1 itself need not be a completely decontextualized or universal statement. It might depend on some contextual aspects that have not yet been specified, and these aspects are to be captured by the context argument. Indeed, it might not be possible ever to completely list all of these context dependencies. At any time, we might have only a partial description of the context; this is why contexts are assumed to be rich objects. In other words, the context object can be thought of as the reification of context dependencies of the sentences associated with the context.

Another way of looking at the context argument is as follows. Imagine a robot which in the course of its duties has to deal with a certain new domain, say Meningitis. In order to do this, it examines the domain, asks people questions, etc., and writes a set of sentences representing this domain. Let us view the action of the robot as a function that computes the representation. The domain itself is of course an argument to this function. In addition, a number of other factors such as the resource constraints on the robot, the parts of the domain that escaped the attention of the robot, the duty the robot was performing that led it to examine the domain, etc., are also influential and hence are arguments to this function. When the robot writes ist(MeningitisContext Rule85), the second arguments to *ist*, i.e., the domain rules of the sort one might find in MYCIN, account for the domain argument while the first argument accounts for the other factors influencing the representation.

In a sense, we are in a position similar to that of the robot (except that we are writing sentences for the computer, not for ourselves) and there are similar factors influencing our representation. The context argument is meant to capture the effect of these factors.

The concept of contexts was first discussed in AI by McCarthy; for a recent reference, see [11]. The idea (that an entity's representation of or knowledge about a subject is influenced by something internal to the entity) has been discussed in philosophy for some time. One philosopher to state this clearly was Husserl [3] (he used the term *Noemata* instead of context), though this theory was severely distorted later in the works of Heidegger [4] and others.

The model theory, proof theory, etc. of the logic of contexts can be found in [8]. Below, we first discuss the structuring of the KB using contexts and then illustrate the kinds of statements that can be made using this logic, by way of a set of examples.

4. Contexts for Structuring the Knowledge Base

In this section, we briefly describe the use of contexts for structuring the Cyc [9] knowledge base. Cyc employs many different kinds of contexts. The most important two of these and their functions are as follows:

Microtheories (MT): A Microtheory contains a set of axioms that constitute a theory of some topic, e.g., a theory of mechanics, a theory of the weather in winter; a theory of what to look for when buying cars, etc. Different microtheories make different assumptions and simplifications about the world with contexts providing a mechanism for recording and reasoning with these assumptions. For any given topic, there may be different microtheories of that topic, at varying levels of detail and generality (possibly mutually contradictory, using different vocabularies, etc.). Microtheories are usually large, long-lived and permanent.

Problem Solving Contexts (PSC): A PSC contains a representation (of some situation) that is tailored for the problem it was set up to solve. E.g., a model of a Christmas tree as a perfect cone, used for determining whether it will fit in a given space in a store window; a model of an object as a point mass for determining its trajectory, etc. A problem solving task might involve answering a single question or a number of related questions. These contexts are usually created dynamically by the system and are ephemeral.

A few other kinds of contexts will be covered in the next section.

Since these contexts are themselves objects in the ontology, the KB also contains descriptions of them, axioms (such as lifting axioms) relating them, etc. These axioms about contexts are used for relative decontextualization, for combining information from different contexts during problem solving, etc. Statements about a context are made in a context that is 'outer' or 'meta' to the above contexts.

There are many advantages of the above structuring of the KB. These are best illustrated by the following examples. The purpose of these examples is twofold. They help to illustrate the kind of statements that can be made using the logic of contexts, and they also show some of the advantages of the context-based structuring of the KB.

(a) We want to state that Gomez Addams (who is very abnormal and always attempts to lose money on investments) is outside the scope of NaiveMoneyMt. To state this, we need the concept of the *Scope* of a theory. We write
¬presentIn(NaiveMoneyMt GomezAddams)
to state Gomez is outside the scope of the

NaiveMoneyMt and
presentIn(NaiveMoneyMt MorticiaAddams)
to state that Morticia Addams is within the scope of NaiveMoneyMt.

If we have a theory of some phenomenon, we would like to maximize its scope, i.e., unless the system knows otherwise, it should assume that a given object is within its scope.

There are two possible intuitive readings of the predicate presentIn. The assertion presentIn(NaiveMoneyMt Joe) could either mean that the term Joe was mentioned explicitly in the NaiveMoneyMt, or that Joe was within the scope/domain of the NaiveMoneyMt (even though the theory does not mention him by name anywhere). We are using presentIn in the latter sense.

(b) Assume we have a category of theories called Naive theories. We want to state that all of them include an explicit assumption that humans are rational. We write this as:

$$(\forall ci\ \mathrm{NaiveTheory}(ci) \Rightarrow \mathrm{ist}(ci\ (\forall\ x\ \mathrm{Human}(x) \Rightarrow \mathrm{Rational}(x))))$$

Here, we are making a statement not just about a particular theory, but about a whole class of theories. Based on some property of the theory (i.e., that it is naive), we are concluding that a certain assertion holds in it.

(c) We might introduce a more sophisticated theory of transactions, GeneralTransactionTheory (GTT), which makes a distinction between different kinds of costs associated with an object. However, we want to use at least some of the axioms in NaiveMoneyMt in GTT and so want to specify the relation between these two theories. We want to state that if the cost of an object X is A in NaiveMoneyMt, the "list price" of X is greater than A in GTT. We write this as,

$$(\forall\ x\ \mathrm{ist}(\mathrm{NaiveMoneyMt}\ \mathrm{cost}(x)=A) \Rightarrow \mathrm{ist}(\mathrm{GTT}\ \mathrm{cost}(x\ \mathrm{ListPrice})>A))$$

There are a few interesting points to note about this example. Firstly, note that these theories use the symbol "cost" in different ways. In GTT it is a binary function and in NaiveMoneyMt it is a unary function.

Also note that based on a statement made about an object (the binding for x) in one context, something may be derived about that object in another context. The two contexts use different vocabularies and make different attributions of an object, but these attributions are about the same object.

The formula in NaiveMoneyMt makes some contextual assumptions. The assumptions behind GTT are different from those of naiveMoneyMt. To compensate for this, the encoding of the proposition 'the price of an object is A' has to be changed when going from one context to another. This process of relative decontextualization is a pragmatic substitute for the complete decontextualization demanded by the traditional model. Rules such as the above are called *Lifting* rules.

(d) We might have more than one theory associated with monetary transactions and these might make mutually inconsistent assumptions. Assume one context (C_1) takes the list price of an object as its cost while another (C_2) includes the sales tax in the cost. We could state this as,

$$\mathrm{ist}(C_1\ \mathrm{cost}(x)=y) \Rightarrow \mathrm{ist}(C_2\ \mathrm{cost}(x)=(*\ y\ 1.07))$$

So it is possible to derive both ist(C_1 cost(g)=\$100) and ist(C_2 cost(g)=\$107). In the previous example, the two contexts involved used different vocabularies. Therefore, different formulas were used to state the same proposition about cost. Here, the two contexts use the same vocabularies, but make different assertions about the cost of the object both refer to as 'g'.

Though the two sentences (cost(g)=\$100) and (cost(g)=\$107) are mutually inconsistent, the KB as a whole should not become inconsistent. We should only be required to maintain local consistency within each context and not necessarily across contexts.

(e) Given a problem for which the NaiveMoneyMt is adequate, we would like to be able somehow to focus attention on this theory and solve the problem using just this theory. Similarly, we should be able to shift focus, enlarge focus, etc.

If the KB contains formulas such as ist(C_1 P_1), ist(C_1 P_2), . . . , ist(C_1 P_n), the system can *enter* C_1 to *be in* C_1. When this is done, the only axioms used by the system are P_1, P_2, . . . , P_n. After problem solving is completed and the system has derived q from P_1, P_2, . . . , P_n, it can *exit* C_1 to get ist(C_1 q).

If the system was asked the query Q or was told P when it was *in* the context C_1, this would be equivalent to exiting C_1 and asking ist(C_1 Q) or asserting ist(C_1 P) respectively.

Not only does the *entering* a context allow the system to focus on a smaller set of axioms, it also allows the system to use an appropriately simpler representation of the domain. For example, for most problems related to commercial transactions, the simpler representation of buying (see section 2.1) would be adequate and the system would not have to deal with the more expressive but also more cumbersome representation of buying involving reification. Not only does this simplify inferencing, but it also makes the task of representation easier. More specifically, assume that the context

C_1 ignores the possibility of simultaneous transactions and hence uses the simpler representation for buying. If the system is *in* C_1, to state that a Joe bought Object001 from Jim we would write occurs((Buying Joe Object001 Jim)) instead of (∃ x Buying(x)∧buyer (Buying001 Joe)∧object (Buying001 Object001)∧seller (Buying001 Jim)).

Entering (and Exiting) a context serve two purposes. One is to provide focus, so as to speed up problem solving behavior. the other is to provide a context for the interaction with the system. The second use is discussed later.

(f) Given a problem, we might not have a context with just the right theory for solving the problem. For example, if the system were asked to determine the least expensive way of getting from Austin to Palo Alto, it might require some axioms from the NaiveMoneyMt and some axioms from the TransportationMt. We should be able to *create* a new context, *lift* the relevant axioms from these two theories into this context, *enter* his context, and solve the problem. Note that these two theories might make different assumptions and use different vocabularies. The lifting process needs to perform the requisite relative decontextualization. In general, this will be far from a complete decontextualization of either of the theories' assertions, as related theories will share many assumptions.

5. Contexts and Natural Language Understanding

The traditional model of natural language understanding (NLU) has been that the NL front end takes natural language utterances as inputs and produces completely decontextualized logical sentences as outputs. As we have claimed repeatedly in the previous sections, completely decontextualized formulas are very difficult to come by, even when a human is manually writing them. Expecting the natural language front end to produce these using only linguistic information (even with the help of some questions the natural language front end asks the KB) might be setting up too difficult a task for it to perform.

The inclusion of contexts into the representation language has a very significant impact on the NLU process. Since the natural language front end is translating into a language in the logic of contexts, many of the decontextualization problems might be handed off to the KB to be dealt with (at a later time).

This changes what it means to translate an NL utterance, i.e., partially decontextualized utterances are now legal formulas in the KB. Understanding an utterance now does not *imply* that it should be completely decontextualized. All it means is that it should be as partially decontextualized as necessary for the use to which the utterance is going to be put. Indeed, it might be possible to return much later to an utterance and decontextualize it a little bit more; this can be done *ad infinitum*.

Reasoning can go on without full decontextualization. As an extreme example, consider the following. The utterance "He is an Engineer" can be translated into the heavily contextdependent formula Engineer(He). In order to answer the question, "Is he a child?" one does not need to know the denotation of 'He'. The translation Engineer(He) is adequate for this purpose and could be used as such to answer this question. There may be other problems for which some decontextualization would be required; when these problems are encountered, a little more decontextualization may be done.

5.1 Understanding Utterances as Constraint Satisfaction

The problem of understanding utterances (multiple sentences or fragments of sentences uttered in particular situations to convey something) is in some ways harder but in many ways easier than understanding sentences considered in isolation.

There are a number of constraints, imposed by the situation in which an utterance is made, on what might possibly be conveyed by a speaker to a listener. In some cases these constraints are enough to narrow down the range of possibilities sufficiently that a simple nod or gesture suffices to convey the intended information. In most cases something more complex is needed, so that a sentence may be uttered. This sentence, in conjunction with the other constraints, conveys the intended information.

Example: Consider the utterance 'Let's talk about the weather in California'. It is ambiguous as to whether 'in California' modifies where the talking should be done or modifies the weather. Now we might happen to know something about the speaker and listener or what they were talking about earlier that might help disambiguate this. If we knew that one of them was thinking of moving to California, we would infer that the phrase 'in California' modifies the weather. On the other hand, if they were planning on meeting in California and were drawing up an agenda of things to do there, and had just been talking about the weather in Alaska, then 'in California' probably modifies the planned talking event.

The problem of understanding an utterance is easier than grappling with a sentence considered in isolation, since there are many constraints that can be used to constrain or bias the information conveyed by the utterance. The problem is harder because of the need to integrate constraints from many different sources. It is difficult to specify *a priori* the sources and types of constraints that may need to be considered. Therefore we need a framework for integrating these different constraints. We already have a tool for dealing with a very wide variety of information, i.e., the knowledge-based system. We will try to adapt this for integrating constraints on the information conveyed by an utterance.

5.2 Framework for Context-based NLU

The overall framework proposed is as follows. We have a hierarchy of contexts based on granularity. At the lowest, most fine-grained level, we have a set of "utterance contexts," one for each utterance. These are ordered based on the temporal ordering of the utterances. Given an utterance, the NL front end creates a new utterance context and asserts the translation of the utterance into this new context.

At the next level, we have a "discourse context" that corresponds to the discourse of which the utterances are a part. The formulas in the utterance context are partially decontextualized and lifted into the discourse context. Depending on the kind of contextual assumptions made, the discourse context might also be a microtheory and/or a problem solving context. Or, it is possible that further decontextualization may have to be done before the contents of the discourse context can be lifted into a microtheory.

In the previous sections we discussed decontextualization of assertions in microtheories and problem solving contexts. Here we discuss some issues associated with the decontextualization of utterance contexts.

We assume that the natural language front end, using the lexicon and purely linguistic knowledge about grammars, morphology, etc., will be able to transform the NL utterance into a formula. This formula might be heavily context dependent, as in Engineer(He). The language use by utterance contexts allows for a number of contextual dependencies. We now briefly examine some of the vocabulary of this language.

(a) Pronouns and Indexicals: *He*, *She*, *It*, *Now*, *I*, etc. are terms in the language.
(b) Definite and indefinite references: The language includes the functions *The* and *A*. The functions *A* and *The* are similar to the articles *A* and The. The sentence "the lady owns a bag" would be translated into owns((The Lady)(*A* Bag)).
(c) We use a variadic function *Etc*. The sentence "Fred likes ice cream, candy, etc." would be translated as likes(Fred (Etc IceCream Candy)).
(d) We have a set of predicates such as *to*, *with*, *for*, etc. The sentence "Fred bought the rose for Jane" would be translated as, (∃ e allInstanceOf(e Buying)∧object(e (The Rose))∧for(e Jane)).
(e) The formulas might explicitly refer to the lexicon. Given the sentence "Joe is at the bank," if the NL front end feels overwhelmed by the number of possible denotations of bank, this sentence may be translated as (exists c at(Joe (The c)) and english Word(c "bank")).
(f) A formula might refer to other utterance contexts. Reference to an utterance context might be used to specify something about the context itself or somehow to qualify a previous utterance. This feature is necessary for translating utterances such as "Going back to the previous topic . . ."

Since the scope of the discourse context is usually much bigger than a single utterance, the language used does not include the above features. Therefore, when lifting a formula from the utterance context to the discourse context, the formula might need to be changed significantly. More specifically, the language of the discourse context will not have any of the earlier mentioned features of the utterance context.

Many different sources of information might affect the translation from utterance context to discourse context. To illustrate this, we now examine some of the different types of heuristics that might be used to determine the referent of 'it' and show sample formalizations of several of these as lifting rules.

5.2.1 Constraints on Resolving 'it'

(a) Linguistic semantic information:
  (i) The denotation of 'it' cannot be a male or female human.

$$(\forall\ c_i\ \mathrm{utteranceContext}(c_i) \Rightarrow \mathrm{ist}(c_i(\forall\ y(=y\ \mathrm{it}) \Rightarrow \neg(\mathrm{male}(y) \vee \mathrm{female}(y)))))$$

(b) Linguistic context information:
  (ii) The denotation of 'it' is an object present in one of the previous contexts.
  (iii) The total number of possible referents of 'it' is to be minimized. We use a slightly more general heuristic and minimize the extents of the utterance contexts; i.e., the utterance contexts use a domain closure assumption.

  Every object referred to in an utterance is put into the domain of the corresponding context. Later we will consider other factors which might require us to add an object to the domain of an utterance context.
  (iv) Given two candidate denotations, the one which occurs in a closer context is preferred. This is an example of something we will be trying to do fairly regularly. Given a set of possible solutions, we want to induce a partial ordering of them and then select the best fit. In this case, the problem for which we are seeking a solution is, (=it ?x).

  There are several deficiencies with the heuristic stating that the reference in the more temporally proximate context is preferred. The most glaring one is that it does not take into account any structure the discourse might have, and instead relies on a purely temporal ordering. Given a set of utterance contexts [uc_1, uc_2, . . , uc_n], in addition to the temporal order, there might be other useful structurings which might be imposed on them. For example, the first three utterances might be about some topic, the next four a digression, and the next three back on the first topic. In such a case, the logical precedent of the eighth utterance is not the seventh one but the third one.

In addition to allPreviousContext, other relations, such as those described in [7] might be used to structure utterance contexts. The structure of the discourse itself may be the topic of an utterance in that discourse. "Cue statements" such as "Going back to . . ." are translated as assertions about the structuring of the utterance contexts.

Introducing a new partial order of contexts (call this one priorContext) is quite easy in this framework (we simply introduce a new relation that can hold between contexts) since contexts are objects in the domain of discourse. However, we have to figure out some way of deducing this new order. We can assume that as a default, priorContext is the same as previousContext. So we have previousContext(c_i c_j) And ¬ab-pc(c_i c_j) ⇒ priorContext(c_i c_j). However, one of the effects of the utterance could be to change what the priorContext of the next utterance is. So (in computer-deutsch) we can have an axiom like (where P denoted the statement—"Going back to . . .")

ist(c_i P)∧contextTopic(c_j X)∧
¬(∃ c_k contextTopic(c_k X)∧
nearer(c_i c_k c_j) ⇒ priorContext(c_i c_j)

(c) Backgrounds of the speaker and listener. This encompasses a very wide variety of heuristics and we shall consider only a simple example. Consider the statement, "The transformer in my amplifier is broken. How do I fix it?" If this statement were being made to a transformer repair person, 'it' might be taken to refer to the transformer. However, if the statement were addressed to the sales representative of the shop where the amplifier was bought, 'it' would probably refer to the amplifier.

(d) The situation of the utterance. This might bring objects into the domain of an utterance context. For example, the speaker and listener are working together to fix a car. The listener has a screwdriver in his hand and the speaker needs it. If the speaker says "Give it to me," the 'it' probably refers to the screwdriver (even though it might never have been explicitly referred to). In general, if a set of people are working together on something, there are objects of shared focus. These, their prominent parts and properties, etc., are included in the domain of the utterance contexts as valid denotations of 'it'.

(e) Felicity constraints. We assume that utterances obey the conversational postulates. For example, as a default, they provide information the listener does not know. Consider the statement "See that bird on the tree—it is made of wood". The 'it' might refer to the bird or the tree. However, the interpretation which assigns the denotation as the tree does not yield any new information. We expect the listener to know that trees are made of wood and the speaker expects the listener to know this. However, since birds are not usually made of wood, the other interpretation, that the bird is made of wood, would indeed provide the listener with new information and this is the preferred interpretation.

(f) Background common sense. This is probably the biggest and most important source of constraints. Let us examine where this fits in the translation process. Consider the utterance, "He brought me a cup of coffee. I drank it quickly." Common sense tells us that 'it' does not refer to the cup, but to the coffee. Let the translation of "I drank it" be P and the corresponding utterance context be c_p. When P is lifted from c_p to the discourse context c_d, linguistic and other constraints identify two possible candidates for the denotation of 'it'—the cup and the coffee. The theory of c_d includes among other things, basic common sense information (only liquids can be drunk, cups are solids, . . .). If we picked the cup as the denotation of 'it', P would be contradictory with the other information in c_d. So the lifting proess eliminates the cup as a candidate and we are left with only possibility of the coffee being the denotation of 'it'.

Common sense may be used not just to eliminate potential interpretations but also to impose a preference ordering on them. For example, consider the utterance, "Mary liked the watch in the shop. She went back and bought it." Based on common sense we know that it is much more likely that Mary would buy the watch than that she would buy the shop. This information can be used to order the likelihood of 'it' as referring to the watch or to the shop.

It is very important to note that we are only aiming at a relative decontextualization, where the extent of the decontextualization is dependent on what the result is going to be used for (i.e., on the context in which the interpretation of the utterance is going to be used).

Another important point is that by using the context framework, we separate out the specification of information that is to be used in understanding an utterance from the procedure to be followed for this understanding. Once the literal, context-dependent translation is obtained from the NL front end, the combination of information from all of the above sources for the purpose of relative decontextualization can be carried out on demand by any inferencing process.

However, there is a downside to viewing the combination of information from different sources as an inferencing process. Since the logic of contexts is very expressive, it is impossible to build a problem solver for performing inferences (with this logic) that is both efficient and guaranteed to be complete. The only option available is to forsake completeness and use a problem solver that handles the most common cases efficiently. We conjecture that a problem solver that was unable to perform complex decontextualizations would be adequate for purposes of understanding most natural language utterances. After all, human problem solvers are far from complete. If an utterance was originally meant to be understood by a human, an incomplete problem solver might be adequate for purposes of decontextualizing it.

6. Comparison to Situation Theory

At first sight, the approach presented here seems similar to that of situation theory (ST). They share the intuition that the content of utterances and mental representations might depend significantly on factors external to the utterance/representation. However, the similarities end there.

In ST, the content of a statement S in a situation c is given by $C_R(S, c)$ [1], where C_R are the *convention constraints* or rules of interpretation. It is (implicitly) assumed that it is possible to define in absolute terms the content of a statement. Indeed, the goal of NL understanding is taken to be the computation of this content. Herein lies the fundamental difference between the approach described in this paper and all previous approaches. In our approach, the statement of the content of an utterance made in a context is itself made with respect to another context, albeit a larger or more general one. The concept of determining the content/meaning of an utterance in absolute terms has been replaced with determining the meaning of an utterance in the context in which this understanding might be used. Along this dimension, our contexts have a much stronger impact on meaning than do situations[1] in situation theory—it does not make sense to talk of the meaning of a statement independent of a context.

Along another dimension, our approach is far less radical than that of situation theory. The model theory (and therefore proof theory) of the logic of contexts, being an extension of the model theory for first order logic, is built using individual objects and set theory. Situation semantics [2] on the other hand, seems to have rejected traditional model theory (and proof theory) as a starting point, and is consequently still at a much earlier stage of development, especially with regard to its proof theory.

7. Conclusion

We began by arguing that all expressions used for representation and communication have contextual dependencies. This context dependency leads to the same fundamental problems in both representation and natural language understanding and begs for a common solution.

To solve some of these problems, we introduced contexts as objects in our ontology. A syntax for recording and reasoning about the context dependency of an expression was introduced. In this syntax a statement is not "universally" true or false, rather, it is true in a context. The use of this logic for solving some of the problems with representation was demostrated by means of a number of examples.

We then focused on the problem of understanding natural language utterances and argued that this is best seen as a constraint resolution problem. We developed a context-based framework for performing an incremental decontextualization of NL utterances by seamlessly integrating constraints from many different sources.

Acknowledgements

We would like to thank Mark Derthick, Karen Pittman, Paul Blair and Mary Shepherd for comments on earlier drafts of this paper.

References

1. BARWISE, J. On the circumstantial relation between meaning and content. In J. Barwise, editor, *The Situation In Logic.* CSLI, Stanford University, CA, 1989.
2. BARWISE, J. Situation theory and related topics. In J. Barwise, editor, *The Situation In Logic.* CSLI, Stanford University, CA, 1989.
3. DREYFUS, H. *Husserl, Intentionality, and Cognitive Science.* MIT Press, Cambridge, 1982.
4. DREYFUS, H. *Being-in-the-world: a commentary on Heidegger's Being and time, division I.* MIT Press, Cambridge, Mass., 1991.
5. FEIGENBAUM, E. A. The art of artificial intelligence. In *Proceedings of IJCAI-77,* Palo Alto, 1977. Morgan Kaufmann.
6. GREINER, R. and LENAT, D. RLL: A representation language language. *In Proc. of the First AAAI Conference* (1980), 165–169.
7. GROSZ, B. and SIDNER, C. The structures of discourse structure. Technical Report CSLI-85-39, CSLI, Stanford University, CA, 1985.
8. GUHA, R. V. Contexts: A Formalization and Applications. Technical Report ACTCYC-423-91, MCC, Austin, TX, 1991.
9. LENAT, D. B. and GUHA, R. V. *Building Large Knowledge Bases.* Addison-Wesley, Reading, Mass., 1990.
10. MCCARTHY, J. Programs with common sense. In Hector Levesque and Ronald Brachman, editors, *Readings In Knowledge Representation.* Morgan Kaufmann, Los Altos, CA, 1986.
11. MCCARTHY, J. Generality in Artificial Intelligence. *Comm. ACM,* 30(12), (1987), 1030–1035.
12. MCCARTHY, J. and HAYES, P. J. Some philosophical problems from the standpoint of AI. In M. Ginsberg, editor, *Readings In Nonmonotonic Reasoning.* Morgan Kaufmann, Los Altos, CA, 1987.
13. QUINE, W. V. *Ontological Relativity and other essays.* Columbia University Press, New York, 1969.
14. QUINE, W. V. *Word and Object.* Columbia University Press, New York, 1969.

(Received November 26, 1991)

[1]Despite their superficial similarities, contexts as described in this paper are very different from situations of situation theory. Situations are real, physical objects. Contexts, on the other hand, are not constrained to be real, and are certainly not physical objects.