Resolving Lexical Ambiguity using a Formal Theory of Context

Saša Buvač
Computer Science Department
Stanford University
Stanford
California 94305-2140
buvac@sail.stanford.edu

June 17, 1995

Abstract

In this paper we show how the formal theory of context, as developed in declarative AI, can be used to represent lexical ambiguity and reason about disambiguation. We first give an introduction to the formal theory of context by briefly sketching its logic, and we identify a new class of contexts, *discourse contexts*. Unlike the previously studied *knowledge base* contexts, which are characterized only by the sentences true in them, discourse contexts are also characterized by the interpretation of their predicate symbols, which can vary from one discourse context to another. We define the notion of ambiguous interpretation and its representation in the language of the context logic, and show how a knowledge base containing general common sense facts can be used to resolve ambiguity. We conclude by showing how a sequence of discourse contexts can be used to represent context change in a question/answer discourse segment, and apply the previously introduced notion of ambiguous interpretation to analyzing the discourse.

1 Introduction

Many words and phrases used in natural language discourse are ambiguous. Their interpretation (and/or meaning) depends on which context they are uttered in. The English word "bank" is one such example. It is ambiguous because it can be interpreted either as a financial institution or a river shore. The choice of interpretation depends on the context in which the word bank is used.[1] This type of ambiguity in natural language is called *lexical ambiguity*; for example, see Hirst 1987 for a discussion of ambiguity in natural language processing.

[1]Note however, that the context is not always sufficient to disambiguate an ambiguous utterance, but is something that needs to be considered in the interpretation of natural language.

Our goal is to be able to write computer programs which will be able to deal with ambiguous expressions. They should be able to disambiguate most ambiguous terms which come up in discourse. The first step in this direction is to provide computers with a language enabling them to declaratively represent that a predicate symbol is ambiguous. Intuitively, this representation will correspond to lexical ambiguity in natural language. The goal of this paper is to provide such a language and show how it can be used in conjunction with a data base of general common sense knowledge to resolve ambiguity.

We will use the logic of context, as developed in declarative AI, to represent and reason about ambiguity. The logic of context provides a language in which contexts are treated as formal objects. The logic extends classical first-order logic by introducing a modality, $\mathtt{ist}(c, \phi)$, used to express that the formula, ϕ, holds in the context c. Using contexts and the `ist` modality we can state that an expression is interpreted differently in different discourse contexts; this serves as the basis for our definition of ambiguous expression. Disambiguating an ambiguous expression involves finding its interpretation in the particular discourse context in which it is used. The above is also expressible in our logic; this enables us to define disambiguation.

The remainder of this paper is organized as follows. In the next section we briefly sketch the logic of context. We then introduce the two types of contexts needed to define ambiguity, knowledge base contexts and discourse contexts, in sections §3. In section §4 we show how ambiguity can be represented in the logic of context. In section §5 a data base of general common sense knowledge is used to reason about and resolve ambiguity. We put together all the concepts developed in the paper to show in section §6 how context change occurring in simple discourse segments is represented and reasoned about. Section §7 contains conclusions and plans for future work.

2 Formal Theory of Context

The logic of context extends classical 2-sorted predicate calculus with identity to enable representing facts about contexts and reasoning with contexts. Our logic has the following four basic features. Firstly, contexts are treated as formal objects, i.e. objects in the semantics which can be denoted by terms in the language and which we can quantify over. Consequently, we can state properties of contexts in the same way we state properties of any other objects. Secondly, rather than being given in isolation, all formulas are stated in some context. We write $\kappa : \phi$ when formula ϕ is given in context κ. Thirdly, the language is extended with a new modality, $\mathtt{ist}(\kappa, \phi)$, (`ist` is pronounced "is true"). It is used to express that the formula, ϕ, is true in the context κ. Fourthly, the formal system contains rules for entering and exiting a context; the proofs which use these rules mirror the intuitive patterns of contextual reasoning. In §4.2 we discuss why these non-classical features of the context logic are useful for capturing ambiguity. In this section we will briefly sketch a basic logic of context which is a trivial generalization of the propositional logic presented in Buvač et al. 1995. Detailed treatment of the subject including soundness, completeness, decidability, and correspondence theorems, as well as a formal comparison of our logic to other modal logics can be found in the latter paper.

2.1 Syntax

A first-order language, $\mathcal{L}$, will be used to express the state of affairs which holds in a particular context as well as the state of a discourse. It is defined in the usual way. If $\mathbb{A}$ is a set of constant symbols, and for each $i \geq 0$ $\mathbb{P}^i$ is a countable set of predicate symbols which take i arguments, then the language $\mathcal{L}$ is a collection of these sets of symbols.

Definition ($\mathcal{L}$): $\mathcal{L} = \mathbb{A} \cup \bigcup_{0 \leq i} \mathbb{P}^i$.

Let $\mathbb{V}$ be the set of variable symbols. We define the set of terms, $\mathbb{T}$, in the usual way to be the union of constant symbols $\mathbb{A}$ and the variable symbols $\mathbb{V}$.

Let $\mathbb{K}$ be the set of all contexts.[2] The set, $\mathbb{W}$, of well-formed formulas (wffs) is the set of classical first-order formulas closed under `ist` modality.

Definition ($\mathbb{W}$): The set $\mathbb{W}$ is the least set satisfying

$$\mathbb{W} = \mathbb{P}^i(\mathbb{T}_1, \ldots, \mathbb{T}_i) \cup (\neg\mathbb{W}) \cup (\mathbb{W} \rightarrow \mathbb{W}) \cup (\forall\, \mathbb{V})(\mathbb{W}) \cup \mathtt{ist}(\mathbb{K}, \mathbb{W})$$

The operations $\exists$, $\wedge$, $\vee$ and $\leftrightarrow$ are defined as abbreviations in the usual way. We adopt the following notational conventions: κ and c will range over $\mathbb{K}$; p will range over $\mathbb{P}$; t will range over $\mathbb{T}$; v will range over $\mathbb{V}$; and lower case Greek letters will range over $\mathbb{W}$.

2.2 Model Theory

Intuitively, a context is modelled by a set of first-order structures that describe what is true in that context. As we will argue later, this will reflect either the possible states of affairs of a knowledge base context or the epistemic state of a discourse context. Thus a model will associate a set of first-order structures with every context. These first-order structures reflect the states of affairs which are possible in a knowledge base context or the epistemic state (i.e. what is known) in a discourse context. For a formula to be true in a context it has to be satisfied by all the first-order structures associated with that context. Therefore the `ist` modality is interpreted

[2] Note that contexts are thus primitive objects of the theory.

as validity: $\mathtt{ist}(\kappa, \rho)$ is true iff the atomic formula ρ is true in all the first-order structures associated with context κ. A system which models a context by a single truth assignment, thus interprets ist as truth, can be obtained by placing simple restrictions on the definition of a model, and enriching the set of axioms. Treatment of ist as validity also corresponds to Guha's proposal for context semantics, Guha 1991, which was motivated by the Cyc knowledge base.

Definition (STR($\mathcal{L}$)): We let STR($\mathcal{L}$) be the set of first-order structures of the language $\mathcal{L}$.
By convention, gothic letters will range over elements of STR($\mathcal{L}$). We use $\mathcal{I}(\mathfrak{A})$ to refer to the interpretation function of the first-order structure $\mathfrak{A}$, and $|\mathfrak{A}|$ to refer to the domain of the structure $\mathfrak{A}$. If X is a set, then $\mathbf{P}(X)$ is the set of subsets of X.

Definition ($\mathfrak{M}$): A *model*, $\mathfrak{M}$, is a function from contexts to sets of first-order structures of some language $\mathcal{L}$,

$$\mathfrak{M} : \mathbb{K} \rightarrow \mathbf{P}(\mathrm{STR}(\mathcal{L})),$$

provided (1) all the contexts have the same domains, i.e. $(\forall \mathfrak{A} \in \mathfrak{M}(\kappa))(\forall \mathfrak{B} \in \mathfrak{M}(\kappa'))|\mathfrak{A}| = |\mathfrak{B}|$; (2) all the domain objects are distinct from the contexts, i.e. the sets $\mathbb{K}$ and $\cup_{\kappa \in \mathbb{K}} \cup_{\mathfrak{A} \in \mathfrak{M}(\kappa)} |\mathfrak{A}|$ are disjoint; and (3) all constants are *rigid designators*, i.e. for any $\mathfrak{A} \in \mathfrak{M}(\kappa_1)$ and $\mathfrak{B} \in \mathfrak{M}(\kappa_2)$, and for any constant a we have $\mathcal{I}(\mathfrak{A})(a) = \mathcal{I}(\mathfrak{B})(a)$.

Definition (variable assignment): A variable assignment is a function from the set of variables, $\mathbb{V}$, to the set of all objects, $|\mathfrak{A}|$ (where $\mathfrak{A} \in \mathfrak{M}(\kappa)$ for any κ). We extend the variable assignment to constants; this is trivial since all the constants are rigid designators.
By convention, the Greek letter σ will range over variable assignments. Instead of writing $\sigma(v)$, we will use the common notation and write $v[\sigma]$.

We introduce $\models$ which is a relation on a model, a first-order structure, a context, a formula, and a variable assignment. It is written

$$\mathfrak{M}, \mathfrak{A} \models \kappa : \phi \, [\sigma],$$

and should be interpreted as a *satisfaction relation*: we say that the model $\mathfrak{M}$, the first-order structure $\mathfrak{A}$, and the variable assignment σ satisfy the formula ϕ in context κ. Note that since every formula is given in a context, the satisfaction relation will also need a context argument.

Definition ($\models$): If $\mathfrak{A} \in \mathfrak{M}(\kappa)$ then $\mathfrak{M}, \mathfrak{A} \models \kappa : \chi \, [\sigma]$ is defined by induction on the structure on χ, as follows:

$\mathfrak{M}, \mathfrak{A} \models \kappa : p^i_j(t_1, \ldots, t_i) \, [\sigma]$ iff $<\mathcal{I}(\mathfrak{A})(t_1[\sigma]), \ \ldots, \ \mathcal{I}(\mathfrak{A})(t_i[\sigma])> \ \in \mathcal{I}(\mathfrak{A})(p^i_j)$

$\mathfrak{M}, \mathfrak{A} \models \kappa : \neg\phi \, [\sigma]$ iff not $\mathfrak{M}, \mathfrak{A} \models \kappa : \phi \, [\sigma]$

$$\mathfrak{M}, \mathfrak{A} \models \kappa : \phi \to \psi\,[\sigma] \quad \text{iff} \quad \mathfrak{M}, \mathfrak{A} \models \kappa : \phi\,[\sigma] \;\Rightarrow\; \mathfrak{M}, \mathfrak{A} \models \kappa : \psi\,[\sigma]$$
$$\mathfrak{M}, \mathfrak{A} \models \kappa : (\forall v)\phi\,[\sigma] \quad \text{iff} \quad \text{for all } d \in |\mathfrak{A}| \quad \mathfrak{M}, \mathfrak{A} \models \kappa : \phi\,[\sigma(v := d)]$$
$$\mathfrak{M}, \mathfrak{A} \models \kappa : \mathtt{ist}(\kappa_1, \phi)\,[\sigma] \quad \text{iff} \quad \text{for all } \mathfrak{B} \in \mathfrak{M}(\kappa_1) \quad \mathfrak{M}, \mathfrak{B} \models \kappa_1 : \phi\,[\sigma]$$

We write $\mathfrak{M}, \mathfrak{A} \models \kappa : \phi$ iff for all σ $\mathfrak{M}, \mathfrak{A} \models \kappa : \phi\,[\sigma]$; and $\mathfrak{M} \models \kappa : \phi$ iff $(\forall \mathfrak{A} \in \mathfrak{M}(\kappa))$ $\mathfrak{M}, \mathfrak{A} \models \kappa : \phi$. Finally, we write $\models \kappa : \phi$ iff $(\forall \mathfrak{M})$ $\mathfrak{M} \models \kappa : \phi$, which corresponds to the provability relation we will define in the formal system.

We conclude the presentation of the model theory with some intuitive clarifications of the satisfaction relation. Firstly, the satisfaction of an atomic formula only depends on the first-order structure, $\mathfrak{A}$, and the variable assignment, σ, in which it is evaluated, and not on the model, $\mathfrak{M}$, and the context, κ. Note, however, that this first-order structure $\mathfrak{A}$ has to be in $\mathfrak{M}(\kappa)$ in order for the satisfaction relation to be defined. Secondly, the set of first-order structures $\mathfrak{M}(\kappa_1)$ reflects the possible states of affairs in context κ_1. Since the $\mathtt{ist}(\kappa_1, \phi)$ is to be interpreted as ϕ being valid in context κ_1, the `ist` clause of the satisfaction relation quantifies over all the first-order structures in $\mathfrak{M}(\kappa_1)$. Thirdly, note that the context κ does not appear in the right hand side of the `ist` clause. Consequently, a context will appear the same regardless which context it is being viewed from. We call this property *flatness*; although it is not desirable in the most general case (see Buvač et al. 1995), flatness is a convenient simplification which we are willing to accept for the purposes of this paper.

2.3 Proof Theory

The formal system contains rules for entering and exiting a context; the proofs which use these rules mirror the intuitive patterns of contextual reasoning. Intuitively, we derive a formula in some context by first entering that context, performing some inferences with the formulas true in that context in order to derive our goal formula, and finally exiting the context. To capture this style of reasoning we define derivability as a relation on a formula ϕ given in a context κ. We also introduce the inference rules (**Enter**) and (**Exit**) which enable the reasoning system to enter and exit a context.

This style of reasoning also has the advantage that at any given point in time the reasoner will only consider the facts which are true in the current context, rather than all the facts given in the knowledge base. The relevance of entering and exiting contexts will become evident in later sections when we give examples of derivations in our logic.

We now present the formal system. Since every formula is given in a context, the provability relation will also need a context argument. Provability is a relation on a context and a wff, which is written

$$\vdash \kappa : \phi$$

and we say that formula ϕ is provable in context κ.

Definition ($\vdash\kappa : \phi$): A formula ϕ *is provable in the context* κ (formally $\vdash\kappa : \phi$) iff $\vdash\kappa : \phi$ is an instance of an axiom schema or follows from provable formulas by one of the inference rules. Formally, iff there is a sequence $[\vdash\kappa_1 : \phi_1, \ldots, \vdash\kappa_n : \phi_n]$ such that $\kappa_n = \kappa$, and $\phi_n = \phi$ and for each $i \leq n$ either $\vdash\kappa_i : \phi_i$ is an axiom, or is derivable from the earlier elements of the sequence via one of the inference rules. In the case of assumptions, formula ϕ is provable from assumptions T in context κ (formally $\mathrm{T}\vdash\kappa : \phi$) iff there are formulas $\phi_1, \ldots, \phi_n \in \mathrm{T}$, such that $\vdash\kappa : (\phi_1 \wedge \cdots \wedge \phi_n) \rightarrow \phi$.
The axiom schemas and inference rules naturally divide into three groups.

1. **Classical Predicate Calculus.**

(PL) $\vdash\kappa : \phi$ provided ϕ is an instance of a propositional tautology

(UI) $\vdash\kappa : (\forall v)\phi(v) \rightarrow \phi(t)$

(MP) $$\frac{\vdash\kappa : \phi \quad \vdash\kappa : \phi \rightarrow \psi}{\vdash\kappa : \psi}$$

(UG) $$\frac{\vdash\kappa : \phi \rightarrow \psi(v)}{\vdash\kappa : \phi \rightarrow (\forall v')\psi(v')}$$ provided v is not free in ϕ

2. **Propositional Properties of Contexts.**

(K) $\vdash\kappa : \mathtt{ist}(\kappa', \phi \rightarrow \psi) \rightarrow (\mathtt{ist}(\kappa', \phi) \rightarrow \mathtt{ist}(\kappa', \psi))$

($\triangle$) $\vdash\kappa : \mathtt{ist}(\kappa_1, \mathtt{ist}(\kappa_2, \phi) \vee \psi) \rightarrow \mathtt{ist}(\kappa_1, \mathtt{ist}(\kappa_2, \phi)) \vee \mathtt{ist}(\kappa_1, \psi)$

(Flat) $\vdash\kappa : \mathtt{ist}(\kappa_2, \mathtt{ist}(\kappa_1, \phi)) \rightarrow \mathtt{ist}(\kappa_1, \phi)$

(Enter) $$\frac{\vdash\kappa' : \mathtt{ist}(\kappa, \phi)}{\vdash\kappa : \phi}$$ (Exit) $$\frac{\vdash\kappa : \phi}{\vdash\kappa' : \mathtt{ist}(\kappa, \phi)}$$

3. **Quantificational Properties of Contexts.**

(BF) $\vdash\kappa : (\forall v)\mathtt{ist}(\kappa', \phi) \rightarrow \mathtt{ist}(\kappa', (\forall v)\phi)$

We briefly comment on the axioms and inference rules. The first set of axioms and rules guarantees that all valid formulas of classical predicate calculus hold in every context, and every context is closed with respect to the classical rules of inference. The second set of axioms and rules captures the propositional modal properties of contextual reasoning. The axiom schema (**K**) guarantees that every context is closed with respect to logical consequence. A property we call *contextual omniscience* is captured by the ($\triangle$) axiom. Intuitively, every context “knows” what is true in every other context. Thus, although a context need not have complete information about what is true in the world, it will have complete information about other context's view of the world. If we interpret contexts as knowledge bases, then contextual omniscience states that every knowledge base "can see" into any other knowledge base. The axiom schema (**Flat**) tells us that every context looks the same regardless of which context it is being viewed from. Although in natural language it is not always the case that only the most deeply embedded context is needed for interpretation, in this paper we restrict our attention to such simpler examples. In Buvač et al. 1995 we present a more general logic which does not make this assumption. Rules (**Enter**) and (**Exit**) allow the formal system to respectively enter and exit a context. Note that the (**Enter**) rule is the converse of the (**Exit**) rule. The third set of axioms and rules captures the quantificational properties of contexts. The Barcan formula (**BF**) states that the domains of all the first-order structures in all of the contexts are the same. Although it will not be desirable in the general case, we assume it for simplicity of presentation.

There are clear parallels between our formal system and the normal systems of modal logic. For example, the rule (**Exit**) is similar to the rule of necessitation in normal systems of modal logic. However, our derivability relation, $\vdash \kappa : \phi$, differs from the usual modal logic derivability relation, $\vdash \phi$. This choice was influenced by the intuition that every formula is given in some context and that the reasoning system can enter and exit a context. Thus when proving that some formula is true in a context, rather than doing all the reasoning in the outermost context (as is done in modal logic), we can enter the indicated context, prove the formula in that context, and exit the context. If we were willing to give up these features we could define derivability in the style standard to modal logics.

Notational Convention: when a formula is given without specifying a context, we are implicitly assuming that it is stated in some outer context $c0$. Formally, when we write ϕ without giving a context, we mean $c0 : \phi$. See McCarthy 1993 for a discussion on the outer context $c0$.

2.4 A Brief Discussion

There is a clear parallel between our logic of context and standard multi-modal logics. In Buvač et al. 1995 we show that the propositional logic of context exactly corresponds to a multi-modal logic in which all modalities satisfy axiom schemas ($\triangle$) and (**Flat**). These results extend trivially to the logic presented in this section. Furthermore, it is straightforward to show that the logic we use in this paper can be reduced to classical first-order logic. Finally, in terms of expressive power, typed lambda calculus, Barendregt 1992, is also at least as powerful as context logic. Although originally typed lambda calculus was not designed for explicitly using multiple contexts, De Bruijn 1991 gives a reduction, referred to as *telescoping*, which enables explicitly representing properties and relations of multiple contexts.

However, apart from being interesting theoretical results, such reductions tell us very little about how to use the formalism, what kind of information should be represented, and for the purposes of this paper: how ambiguity should be represented and resolved. For example, we can take this line of reasoning a step further and reduce all of the above mentioned formalisms to Turing machines. But this does not make Turing machines the only interesting formalism. Instead, it demonstrates that a particular formalism is useful if and only if its semantics are appealing, i.e. if it allows us to represent facts about the world and the domain of study in a simple, concise and intuitive fashion.

3 Two Types of Contexts

In this section we identify two types of contexts: the knowledge base contexts and discourse contexts. The former have been widely used in AI. The latter need to be introduced in order to represent and reason about ambiguity and context change.

3.1 Knowledge Base Contexts

Knowledge base (kb) contexts reflect some possible states of affairs. By "possible" we mean that a knowledge base context need not completely specify what holds in the world, i.e. it might "not be committed" about some propositions. For example, the context of a common sense knowledge base of Stanford University might contain the proposition "kids drive BMW's" reflecting on the states of affairs at Stanford in which kids drive BMW's. The set of first-order structures which a model associates with a kb context reflect this possible states of affairs of that context. The modality $\mathtt{ist}(\kappa, \phi)$ expresses that the formula ϕ holds in all the possible states of affairs described by the kb context κ. In the above example, we would say $\mathtt{ist}(\text{Stanford}, \text{kids drive BMW's})$. This clarifies the need for the universal quantification in the `ist` clause in the satisfaction relation. A model, $\mathfrak{M}$, satisfies the formula $\mathtt{ist}(\text{Stanford}, \text{kids drive BMW's})$ iff the proposition "kids drive BMW's" holds in all the possible states of affairs that $\mathfrak{M}$ associates with the context of Stanford. In other words, if all the first-order structures $\mathfrak{A} \in \mathfrak{M}(\text{Stanford})$ satisfy the proposition "kids drive BMW's". Formally, we can write

$$\mathfrak{M} \models \text{Stanford} : \text{kids drive BMW's}$$

which by the `ist` clause of the satisfaction relation can be written as

$$(\forall \mathfrak{A} \in \mathfrak{M}(\text{Stanford})) \quad \mathfrak{M}, \mathfrak{A} \models \text{Stanford} : \text{kids drive BMW's}$$

and naively translating the English sentence "kids drive BMW's" into logic enables us to apply the atomic clause of the satisfaction relation to obtain

$$(\forall \mathfrak{A} \in \mathfrak{M}(\text{Stanford})) \quad <\text{kids}, \text{BMW's}> \in \mathcal{I}(\mathfrak{A})(\text{drive}).$$

In the above derivation we have disregarded variable assignments since only constants appeared as terms in our formula. Finally, note that in this paper we are assuming that all the contexts have the same domains. Thus, contrary to one's intuitions, the context of Stanford does not determine the domain of discourse to contain all the objects in Stanford as elements. All the objects are present in all contexts and the context only determines what relations hold between these objects. By convention, c_{kb} will be used to refer to knowledge base contexts.

The problem of generality was the primary motivation for the use of contexts in AI. KB contexts are central for dealing with this problem. A simple manifestation of the problem of generality is that AI systems cannot coherently use knowledge bases which were not originally intended to be used together, which in fact might, on the surface, seem inconsistent. For example, assume a common sense knowledge base of Stanford University contains the proposition "kids drive BMW's". A common sense knowledge base of Berkeley, which was not originally intended to be used with the above mentioned Stanford knowledge base, will probably contain the negation of this proposition.[3] A logic dealing with kb contexts enables a reasoning system to use such seemingly inconsistent knowledge bases without deriving a contradiction. Every knowledge base will be given in a separate context and *lifting axioms* will be used to relate the two contexts. For a detailed exposition of this style of reasoning see McCarthy 1993, McCarthy and Buvač 1994.

3.2 Discourse Contexts

To represent and reason about ambiguity in the framework of the formal theory of context, we identify a new class of contexts, the *discourse contexts*. Previously studied *knowledge base* contexts are characterized only by the sentences which are true in them. Discourse contexts, however, are not only characterized by the sentences which are true in them but also by the interpretation of their predicate symbols, which might vary from one discourse context to the next.

Discourse contexts are used to represent a particular state in a discourse. In our simple model, the discourse state consists of the

epistemic state consisting of the facts established in a discourse or facts which are known based on the discourse, and the

semantic state describing the interpretation of predicate symbols in the discourse context by relating them to predicate symbols in some kb context.

By convention, c_{d} will refer to a discourse context.

[3] A note for non-US audience: this example makes use of University pop culture according to which Stanford students, unlike Berkeley students, are often stereotyped to be spoiled children who drive expensive cars.

To illustrate the notion of the discourse state, we examine a simple discourse context c_{d}, resulting in me uttering a single propositional sentence "Vanja is running" to my mother. The epistemic state of c_{d} will contain (only) the fact that Vanja is running. The semantic state of c_{d} describes the meaning of predicate symbols "being named Vanja" and "running". Since I was talking to my mother the intended interpretation of "Vanja" is my brother Vanja, and since Vanja is an exercise freak, the intended interpretation of running is the standard athletic one. Formally, the semantic state of c_{d} will be a relation between the predicate symbol "Vanja" (or "being named Vanja") in c_{d} and a predicate symbol "is-Saša-Buvač's-only-brother" in some knowledge base context containing facts about my family. It will also include the relation between the predicate symbol "running" in the discourse context c_{d} and the predicate symbol "running" in some knowledge base context containing descriptions of exercising. Later we will give more elaborate examples of how our formalism can be used to represent discourse states.

The epistemic state of a discourse context is similar to the state of affairs which is captured by kb contexts. In the model theory, the first-order structures which a model associates to a discourse context capture the epistemic state of the discourse context, i.e. what is known to be said in the discourse. In the logic, we talk about the epistemic state of a discourse context using the `ist` modality. In the above example, we would express the epistemic state of c_{d} by saying $\mathtt{ist}(c_{\mathrm{d}}, \text{Vanja is running})$.

The semantic state of a discourse context captures the interpretation of predicate symbols used to express the epistemic state of the discourse context, i.e. the *interpretation of discourse predicate symbols.* This is done by relating the truth of predicate symbols in the discourse context to the truth of formulas in some kb context. This can also be represented in the basic context language using the `ist` modality.

Definition (interpreting discourse predicate symbols): Predicate symbol p in the discourse context c_{d} is interpreted as ϕ in the kb context c_{kb} if

$$\mathtt{ist}(c_{\mathrm{d}}, p(\vec{x})) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \phi(\vec{x}))$$

As is common in logic, we assume that free variables are implicitly universally quantified. Note that the semantic state is an intuitive notion which is not captured directly by the formalization. Unlike the epistemic state of a context which is captured by the set of first-order structures associated with that context, we can not point to a formal construct and say "look, this is the semantic state." The above formula expresses a relation between a discourse context and a kb context. In other words, the formula describes the semantic state only given a context of a knowledge base.

Although the above formula defines a very simple, extensional relation between p and ϕ, it is sufficiently expressive to capture all the notions we are currently working on. Returning to the above example, to give the interpretation of the discourse predicate symbol "Vanja" we would say

$$\mathtt{ist}(c_{\mathrm{d}}, \mathrm{Vanja}(x)) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \text{is-Saša-Buvač's-only-brother}(x))$$

and the interpretation of the discourse predicate symbol "run" would be given by

$$\mathtt{ist}(c_{\mathrm{d}}, \mathrm{run}(x)) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \mathrm{run}(x))$$

where c_{kb} is the context of some knowledge base containing facts both about athletics and about my family.

Since we are not concerned with solving the syntactic and semantic problems addressed by the natural language community, we are assuming the system is given the discourse utterances in the form of logical formulas. This assumption is in line with McCarthy 1990a; in McCarthy's terminology we would say that the discourse has been processed by both the parser and the understander to produce a logical theory. Note that the process of producing this theory is not precisely defined, and it is not completely clear how much common sense information is needed to generate it. It might turn out that producing such a theory requires the solution of the problem we had set out to solve. But for the time being let us take a positive perspective and assume the discourse theory is given.

To this point, discourse contexts have not been studied in formal theories of context because they are not as central to the problem of generality as the kb contexts. However, they seem relevant to AI as tools for analyzing discourse as well as a canonical example of contexts associated with particular circumstances. Although in later sections we will only deal with question-answer discourses (i.e. discourses consisting of a sequence of questions and answers), at this point we consider discourse in the general sense.

Note that our simple model does not claim to capture all aspects of discourse interpretation. We have refrained from modeling some phenomena that have been studied by semanticists and computational linguists. In particular, Discourse Representation Theory, Kamp 1981, includes a third aspect of discourse interpretation, namely discourse entities known as *reference markers*. Reference markers, each of which can be accessible at a given point in a discourse, are now viewed as an essential element of most theories of context that deal with anaphoric reference. Furthermore, we have ignored pragmatic aspects relevant to discourse analysis; see Grosz and Sidner 1986. These include resolving references by keeping track of which objects are salient in a discourse, and inferring the intentions of agents based on their speech acts. Extending the notion of a discourse state to deal with these and other linguistic phenomena is part our plans for future research.

4 Representing Ambiguity

The same predicate symbol will often be interpreted differently in different discourse contexts. For example, in some discourse contexts the predicate symbol *bank* will be interpreted as a shore of a river and in other discourse contexts it will be interpreted as a financial institution. We say that such predicate symbols are ambiguous.

Ambiguity is a property of the semantic states of discourse contexts. First, we introduce the notion of unambiguous interpretation.

Definition (unambiguous interpretation): Predicate symbol p is interpreted unambiguously as ϕ in the kb context c_{kb}, if p in every discourse context c_{d} is interpreted as ϕ in c_{kb}.
Now we define ambiguous interpretation.

Definition (ambiguous interpretation): Predicate symbol p is interpreted ambiguously as ϕ or as ϕ' in the kb context c_{kb}, if p in every discourse context c_{d} is interpreted as ϕ in c_{kb} or is interpreted as ϕ' in c_{kb}; formally

$$(\forall\vec{x})(\mathtt{ist}(c_{\mathrm{d}}, p(\vec{x})) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \phi(\vec{x})))\vee$$
$$\vee(\forall\vec{x})(\mathtt{ist}(c_{\mathrm{d}}, p(\vec{x})) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \phi'(\vec{x})))$$

where ϕ and ϕ' are not logically equivalent. Note that in the above definition an ambiguous interpretation can be ambiguous in only two different ways. The definition generalizes in a natural way for allowing an arbitrary number of ambiguities, although as long as only lexical ambiguities are concerned the number is likely to be small for the cases that actually come up in natural language.

Finally, we introduce the notion of disambiguating.

Definition (disambiguating): Disambiguating is determining the interpretation of an ambiguous predicate symbol p in a particular discourse context c_{d}.

Our notion of ambiguous interpretation is defined in terms of predicate symbols in a logical language. However, when one thinks of ambiguous interpretation one is usually concerned with natural language utterances. If we assume some *prima facia* translation of natural language into logic, ambiguous interpretation as defined above will correspond closely to *lexical ambiguity*. This will be illustrated with a concrete example in section §4.1.

Our formalism treats ambiguity as a disjunction of the `ist` modalities. This should be distinguished from the treatment of ambiguity as disjunction in classical logic. We discuss this issue further after giving an example.

4.1 The Bank Example

The predicate symbol *bank* is ambiguous. It can be used to either refer to a financial institution or to a river shore. The choice of interpretation depends on the discourse context in which the word bank is used. This ambiguity is an example of homonymy; in section §6.3.3 we give an example of ambiguity which is based on polysemy.

Assume a knowledge base is given in a context c_{kb}. Also assume the predicate symbol *at* is unambiguous

$$(1) \qquad (\forall x, y)\mathtt{ist}(c_{\mathrm{kb}}, at(x, y)) \leftrightarrow \mathtt{ist}(c_{\mathrm{d}}, at(x, y)).$$

Since the predicate symbol *bank* is ambiguous we have the axiom

$$(2) \qquad \begin{aligned} &(\forall x)(\mathtt{ist}(c_{\mathrm{kb}}, \textit{river-bank}(x)) \leftrightarrow \mathtt{ist}(c_{\mathrm{d}}, \textit{bank}(x)))\vee \\ &(\forall x)(\mathtt{ist}(c_{\mathrm{kb}}, \textit{financial-bank}(x)) \leftrightarrow \mathtt{ist}(c_{\mathrm{d}}, \textit{bank}(x))) \end{aligned}$$

The utterance "Vanja is at a bank" might result in a particular discourse context c_{d} containing the sentence

$$(3) \qquad c_{\mathrm{d}} : \quad (\exists x)\textit{bank}(x) \wedge at(\textit{Vanja}, x).$$

Therefore, by exiting context c_{d}, the following is true in the knowledge base:

$$(4) \qquad \mathtt{ist}(c_{\mathrm{d}}, (\exists x)\textit{bank}(x) \wedge at(\textit{Vanja}, x)).$$

Note that this defines the epistemic state of the context c_{d}. Since the domains of all contexts are the same, which is given by the (**BF**) axiom, we get

$$(5) \qquad (\exists x)\mathtt{ist}(c_{\mathrm{d}}, \textit{bank}(x) \wedge at(\textit{Vanja}, x)).$$

Using the fact that `ist` distributes through a conjunction, which follows trivially from the axiom schema (**K**), we have

$$(6) \qquad (\exists x)\mathtt{ist}(c_{\mathrm{d}}, \textit{bank}(x)) \wedge \mathtt{ist}(c_{\mathrm{d}}, at(\textit{Vanja}, x)).$$

Since the *at* predicate symbol is unambiguous, using axiom 1 we get

$$(7) \qquad (\exists x)\mathtt{ist}(c_{\mathrm{d}}, \textit{bank}(x)) \wedge \mathtt{ist}(c_{\mathrm{kb}}, at(\textit{Vanja}, x)).$$

Using the ambiguity of the *bank* predicate symbol given in axiom 2 we get

$$(8) \qquad \begin{aligned} &(\exists x)(\mathtt{ist}(c_{\mathrm{kb}}, \textit{river-bank}(x)) \vee \mathtt{ist}(c_{\mathrm{kb}}, \textit{financial-bank}(x)))\wedge \\ &\wedge \mathtt{ist}(c_{\mathrm{kb}}, at(\textit{Vanja}, x)). \end{aligned}$$

Thus we have shown that the unambiguous interpretation of the predicate symbol *at* and the ambiguous interpretation of the predicate symbol *bank* allows us to conclude in the kb context that Vanja is either at a river bank or at a financial bank, but we do not know which.

4.2 Treating Ambiguity as Disjunction

Naïvely, ambiguity could be treated as a classical disjunction. In this light, the fact that the predicate symbol *bank* is ambiguous would be given by

(9) $(\forall x)\mathtt{ist}(c_{\mathrm{kb}}, \mathit{river\text{-}bank}(x) \vee \mathit{financial\text{-}bank}(x)) \leftrightarrow \mathtt{ist}(c_{\mathrm{d}}, \mathit{bank}(x))$

and the utterance "Vanja is at a bank", given in formula 3, would allow us to derive

(10) $(\exists x)(\mathtt{ist}(c_{\mathrm{kb}}, (\mathit{river\text{-}bank}(x) \vee \mathit{financial\text{-}bank}(x)) \wedge \mathit{at}(\mathit{Vanja}, x)))$.

We feel that the treatment of ambiguity using classical disjunction is inadequate. Intuitively, in the above example, it is not the case that we are communicating that Vanja is either at the river bank or at the money bank. Instead, it is the case that we are either communicating that Vanja is at the river bank, or we are communicating that Vanja is at the money bank. Therefore, associated to the knowledge base context in which the utterance "Vanja is at a bank" is interpreted should not be two first-order structures, one where Vanja is at the river bank and another where Vanja is at the money bank. Instead, there should be a single first-order structure, either the one in which Vanja is at the river bank or the one in which Vanja is at the money bank. Thus we advocate formula 8 over formula 10 as the correct reading of "Vanja is at a bank." Formally, formula 8 is logically stronger than 10 because 8 requires that all the first order structures interpret bank as river bank or all the first order structures interpret bank as financial bank, whereas 10 does not require this unanimity. Since this reading which we are trying to capture is not expressible in classical first-order logic (in a straightforward fashion) we are proposing the use of context logic to represent ambiguity.

5 Disambiguating via Common Sense Knowledge

In this section we illustrate how the context formalism makes it simple, in principle, to use general common sense knowledge to resolve ambiguity. The utility of this approach will be amplified

The necessity to use common sense knowledge in natural language processing is now widely acknowledged, and has recently been argued for in detail in Kameyama 1994. Contrasting Kameyama's examples with the example that follows should convince the reader of advantages the context formalism has (at least in terms of elegance).

The use of common sense knowledge is one of many possible ways of resolving ambiguity. In section §6.3.3 disambiguation is performed by directly asking the speaker which particular meaning of the word was desired. This, "direct asking", approach to disambiguation is significantly simpler since no common sense knowledge is needed, but it requires more logical machinery (to represent the sequence of utterances which make up a discourse). What we will probably want in the long run is a combination of the two approaches. The "obvious" ambiguities will be resolved using common sense knowledge, and where that fails the computer program will go back to the user and ask him or her to clarify what they said.

Now we use an example to show how common sense knowledge can be used to resolve ambiguity. We will conclude this section by explicating some assumptions underlying our approach and comparing our form of disambiguation to existing linguistic theories.

5.1 Disambiguating the Bank Example

The utterance which we examined previously, "Vanja is at a bank", can not be disambiguated without additional information. Thus we now turn our attention to the utterance "Vanja is getting money at a bank", in which "bank" should intuitively refer to a financial bank. Assume it is uttered in some discourse context c_{d}. We thus have

(11) $$c_{\mathrm{d}}: \quad (\exists x) \mathit{bank}(x) \wedge \mathit{get}(\mathit{money}, \mathit{Vanja}, x).$$

A derivation similar to the one in section §4.1 will give us

(12) $$(\exists x)(\mathtt{ist}(c_{\mathrm{kb}}, \mathit{river\text{-}bank}(x)) \vee \mathtt{ist}(c_{\mathrm{kb}}, \mathit{financial\text{-}bank}(x))) \wedge$$
$$\wedge \mathtt{ist}(c_{\mathrm{kb}}, \mathit{get}(\mathit{money}, \mathit{Vanja}, x)).$$

Assume a common sense knowledge base, $C_{\text{common-sense}}$, contains general common sense facts, including:

(13) $$C_{\text{common-sense}}: \quad \mathit{get}(\mathit{money}, x, y) \rightarrow \mathit{financial\text{-}institution}(y)$$

(14) $$C_{\text{common-sense}}: \quad \mathit{river\text{-}bank}(x) \rightarrow \neg \mathit{financial\text{-}institution}(x).$$

In order to use this information, we will need to *lift* it into our knowledge base:

(15) $$\mathtt{ist}(C_{\text{common-sense}}, \phi) \rightarrow \mathtt{ist}(c_{\mathrm{kb}}, \phi)$$

For more examples and discussion on lifting see McCarthy and Buvač 1994. The above formulas 12–15 allow us to disambiguate formula 11 and infer

(16) $$(\exists x)\mathtt{ist}(c_{\mathrm{kb}}, \mathit{financial\text{-}bank}(x) \wedge \mathit{get}(\mathit{money}, \mathit{Vanja}, x))$$

telling us that Vanja is getting money at a financial bank. The derivation is straightforward.

5.2 Some Underlying Assumptions

We point to two assumptions which need to be made in order for our approach to be useful in practice.

Once we are given a particular instance of an ambiguous expression it is easy to write a few common sense axioms which will be used to disambiguate it. However, the reader might be skeptical about the existence of a large common sense knowledge base which contains enough information to disambiguate most expressions that come up in natural language discourse and were not anticipated in advance. This is in fact a valid concern, and the practical utility of our approach to disambiguation is contingent on the existence of such a knowledge base. In our defense we point to (1) active research in logical AI, the goal of which is creating such knowledge bases of common sense facts; and (2) the Cyc system, Guha and Lenat 1990, which is an existing large common sense knowledge base of facts usable in various arbitrary circumstances.

Our solution also hinges on the assumption that all common sense knowledge is shared by all speakers, and thus can be lifted not only into the kb context, but also into the context of what we know about the knowledge of the speaker. When I say "Vanja is goes to the bank because he needs money," my information about Vanja's knowledge is what counts. We consider this assumption to be reasonable because all participants in natural language discourse ought to have common sense (which is why it is called "common" sense). If the reader is not comfortable with this assumption then our approach can be viewed as a useful approximation.

5.3 Comparison to Existing Approaches

Many linguistic theories are concerned with resolving ambiguity. Probably the two best known formal frameworks are *discourse representation theory*, Kamp 1981, and *typed lambda calculus*, Barendregt 1992. Since the latter is more general than the former, in the sense that the structures of discourse representation theory are embeddable in expressions of typed lambda calculus (see Ahn and Kolb 1990), we will compare our disambiguation approach to the one commonly used in typed lambda calculus.

A naïve system that uses types for lexical disambiguation might handle the disambiguation problem presented in section §5.1 in the following fashion. We would have types *location* and *financial-institution* for objects that are locations or financial institutions, respectively. We assume that river banks are of type *location* but not of type *financial-institution*. The phrase "getting money" would take an object of type *financial-institution*. Consequently, "Vanja is getting money at a bank" could not mean that Vanja is getting money at a river shore but only that Vanja is getting money at a financial bank.

Using the formal theory of context has the advantage over type-based disambiguation strategies like the one described above. It makes it simple to use general common sense knowledge which is stated in full first-order logic rather than having to be encoded consequences of common sense facts in terms of types of arguments. To illustrate this point we will introduce some pragmatic issues thus making the above example slightly more complicated. Assume that the ambiguous sentence was "Vanja is going to the bank because he needs money." Common sense knowledge could be used to disambiguate "bank" in the following fashion. We are told that Vanja needs money. It follows that he has the goal of getting money. Therefore he is performing actions to achieve the goal of getting money. Going to a financial bank is a way of achieving the goal, where as going to a river bank is not. Thus "bank" will refer to a financial bank. Formalization of goals and actions needed to achieve them is the kind of information that should be found in common sense knowledge bases. Encoding facts about goals and actions in form of types does not seem natural.

6 Context Change in a Discourse Segment

In previous sections we have examined discourse contexts in isolation. We have assumed that a particular discourse context was given, and we focused on characterizing it and relating it to some kb context. Now we focus on context change: we will use discourse contexts to represent and reason about the change that takes place in simple sequences of questions and answers.

We examine question/answer discourse segments which are simply sequences of questions and answers. In this simple model we allow two types of questions:

propositional questions are used to inquire whether a proposition is true or false; they require a yes or no answer. In the language we introduce a special proposition *yes* which is used to answer these questions. In linguistics these are commonly referred to as YN questions.

qualitative questions are used to find the objects of which a formula holds; in the language we introduce a unary predicate symbol *answer* which holds of these objects. In linguistics these are commonly referred to as WH questions.

In order to know what is being communicated in a discourse, as well as reason about a discourse in general, we need a way of representing the discourse. We represent a discourse segment with a sequence of discourse contexts, each of which in turn represents the discourse state after an utterance in the discourse. Our attention is focused only on discourse segments which are sequences of questions and replies: $[q^1, r^1, q^2, r^2, \ldots, q^n, r^n]$. Thus, we can represent such a discourse segment with a sequence of discourse contexts:

$$[c_{\mathrm{d}}, \mathtt{query}(c_{\mathrm{d}}, q^1), \mathtt{reply}(\mathtt{query}(c_{\mathrm{d}}, q^1), r^1), \ldots$$
$$\ldots, \mathtt{reply}(\mathtt{query}(\mathtt{reply}(\cdots \mathtt{reply}(\mathtt{query}(c_{\mathrm{d}}, q^1), r^1) \cdots, r^{n-1}), q^n), r^n)]$$

s.t. (i) c_{d} is some discourse context in which the initial question (q^1) was asked; (ii) the function `query` takes a question ϕ and some discourse context c_{d} (representing the discourse state before the question ϕ) and returns the discourse context representing the discourse state after asking the question ϕ in c_{d}; (iii) the function `reply` takes a reply ϕ and some discourse context c_{d} (representing the discourse state before replying ϕ) and returns the discourse context representing the discourse state after replying ϕ in c_{d}. In order to reason about the discourse segment we now only need the properties of the functions `query` and `reply`.

6.1 The Logic of `query` and `reply`

In this section we give the properties of the functions `query` and `reply`, which are central for representing question/answer discourse segments.

For simplicity of presentation we will not change our basic logic. Thus the `query` and `reply` functions will not be in the language of the logic, but at the meta-level. Consequently the axioms that the two functions admit, which are given below, are meant as meta-syntaxis and could be referred to as *meaning postulates* in the style of meaning postulates of Montague Grammar. Extending the basic logic of context to incorporate `query` and `reply` as logical functions is straightforward.

We begin with some intuitions. Intuitively, the `query` function will set up a context in which the reply to the question will be interpreted. For example, the context resulting in asking some proposition p will have the property that *yes* in that context will be interpreted as p. Thus `query` only changes the semantic state of the discourse context. The `reply` function will do a simple update of information: the formulas true in the context resulting in replying p in c_{d} will be exactly those formulas which are conditionally true on p in c_{d}. Thus the `reply` function only changes the epistemic state of the discourse context. We now make these notions more precise.

The following axioms characterize the functions `query` and `reply`.

interpretation axiom (propositional) if ϕ is a closed formula, then

$$\mathtt{ist}(\mathtt{query}(\kappa, \phi), \phi \leftrightarrow \mathit{yes})$$

frame axiom (propositional) if ϕ is a closed formula, and *yes* does not occur in ψ, then

$$\mathtt{ist}(\kappa, \psi) \rightarrow \mathtt{ist}(\mathtt{query}(\kappa, \phi), \psi)$$

interpretation axiom (qualitative) if x is the only variable occurring free in ϕ, then

$$\mathtt{ist}(\mathtt{query}(\kappa, \phi(x)), \phi(x) \leftrightarrow \mathit{answer}(x))$$

frame axiom (qualitative) if x is the only variable occurring free in ϕ, and *answer* does not occur in ψ, then

$$\texttt{ist}(\kappa, \psi) \rightarrow \texttt{ist}(\texttt{query}(\kappa, \phi(x)), \psi)$$

reply axiom

$$\texttt{ist}(\texttt{reply}(\kappa, \phi), \psi) \leftrightarrow \texttt{ist}(\kappa, \phi \rightarrow \psi)$$

Assume that *no* is defined as an abbreviation for $\neg yes$. We proceed to illustrate the axioms and their use with an example.

6.2 Example: Air Force–GE Discourse

We examine the following hypothetical discourse taking place between the Air Force and General Electric:

1. AF: Will you bid on the engine for the FX22?
2. GE: Yes.
3. AF: What is your bid?
4. GE: \$2M.
5. AF: Does that include spares?
6. GE: Yes.

We transcribe the above discourse segment in our logic as a sequence of discourse contexts, s.t.

$c1 = \texttt{query}(c, \textit{will-bid-on}(\text{FX22-engine}))$
$c2 = \texttt{reply}(c1, \textit{yes})$
$c3 = \texttt{query}(c2, \textit{price}(\text{FX22-engine}, x))$
$c4 = \texttt{reply}(c3, \textit{answer}(\$2M))$
$c5 = \texttt{query}(c4, \textit{price}(x) \leftrightarrow \texttt{ist}(c_{\text{kb}}, \textit{price-including-spares}(x)))$
$c6 = \texttt{reply}(c5, \textit{yes})$

where c is the initial discourse context.

6.3 Deriving Properties of the Air Force–GE Discourse

We now show some properties of the discourse segment which can be derived with our logic.

6.3.1 First Question: Propositional Case

The discourse segment begins with a propositional question. We show how the first two utterances modify the discourse state.

Proposition ($c2$): $\texttt{ist}(c2, \textit{will-bid-on}(\text{FX22-engine}))$

Proof ($c2$): Instantiating the first axiom for the propositional questions, we get

$$\texttt{ist}(\texttt{query}(c, \textit{will-bid-on}(\text{FX22-engine})), \textit{will-bid-on}(\text{FX22-engine}) \leftrightarrow \textit{yes}) \quad (17)$$

which, by definition of $c1$, can be written as

(18) $$\mathtt{ist}(c1, \textit{will-bid-on}(\text{FX22-engine}) \leftrightarrow \textit{yes})$$

Instantiating the axiom for reply we have

(19) $$\mathtt{ist}(\mathtt{reply}(c1, \textit{yes}), \textit{will-bid-on}(\text{FX22-engine})) \leftrightarrow$$
$$\leftrightarrow \mathtt{ist}(c1, \textit{yes} \rightarrow \textit{will-bid-on}(\text{FX22-engine}))$$

and it follows from the two lines above that

(20) $$\mathtt{ist}(\mathtt{reply}(c1, \textit{yes}), \textit{will-bid-on}(\text{FX22-engine}))$$

which by definition of $c2$ we can write as

(21) $$\mathtt{ist}(c2, \textit{will-bid-on}(\text{FX22-engine}))$$

$\square_{\mathbf{c2}}$

6.3.2 Second Question: Qualitative Case

The reasoning for this qualitative question is similar to the propositional question.

Proposition ($c4$): $\mathtt{ist}(c4, \textit{price}(\text{FX22-engine}, \$2M))$

Proof ($c4$): We begin with an instance of the first axiom for qualitative questions

(22) $$\mathtt{ist}(\mathtt{query}(c2, \textit{price}(\text{FX22-engine}, x)),$$
$$\textit{price}(\text{FX22-engine}, x) \leftrightarrow \textit{answer}(x))$$

which, by definition of $c3$, can be written as

(23) $$\mathtt{ist}(c3, \textit{price}(\text{FX22-engine}, x) \leftrightarrow \textit{answer}(x))$$

Instantiating the axiom for reply we have

(24) $$\mathtt{ist}(\mathtt{reply}(c3, \textit{answer}(\$2M)), \textit{price}(\text{FX22-engine}, \$2M)) \leftrightarrow$$
$$\leftrightarrow \mathtt{ist}(c3, \textit{answer}(\$2M) \rightarrow \textit{price}(\text{FX22-engine}, \$2M))$$

and it follows from the two lines above that

(25) $$\mathtt{ist}(\mathtt{reply}(c3, \textit{answer}(\$2M)), \textit{price}(\text{FX22-engine}, \$2M))$$

which by definition of $c4$ we can write as

(26) $$\mathtt{ist}(c4, \textit{price}(\text{FX22-engine}, \$2M))$$

$\square_{\mathbf{c4}}$

Due to the frame axioms, the conclusion established in the first question, i.e. given in formula 21, also holds in context $c4$.

Proposition (frame): $\texttt{ist}(c2, \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$

Proof (frame): We first instantiate the second axiom for qualitative questions to get

$$(27)\qquad \texttt{ist}(c2, \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine})) \rightarrow$$
$$\rightarrow \texttt{ist}(\texttt{query}(c2, \mathit{price}(\text{FX22-engine}, x)), \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

The two lines above imply

$$(28)\quad \texttt{ist}(\texttt{query}(c2, \mathit{price}(\text{FX22-engine}, x)), \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

which, by definition of $c3$, can be written as

$$(29)\qquad \texttt{ist}(c3, \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

Now we apply the following instance of the reply axiom

$$(30)\qquad \texttt{ist}(\texttt{reply}(c3, \mathit{answer}(\$2M)), \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine})) \leftrightarrow$$
$$\leftrightarrow \texttt{ist}(c3, \mathit{answer}(\$2M) \rightarrow \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

to get

$$(31)\qquad \texttt{ist}(\texttt{reply}(c3, \mathit{answer}(\$2M)), \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

which, by definition of $c4$, can be written as

$$(32)\qquad \texttt{ist}(c4, \mathit{will\text{-}bid\text{-}on}(\text{FX22-engine}))$$

$\Box$**frame**

6.3.3 Third Question: Dealing with Ambiguity

We are assuming that the predicate symbol *price* is ambiguous in the discourse contexts since it can be interpreted as either *price-including-spares* or as *price-not-including-spares* in some knowledge base. Note that although this is still a lexical ambiguity, it is an example of polysemy, rather than of homonymy as was the case with "bank". In the third question the predicate symbol is disambiguated for context $c6$. Since in the setting of this discourse it was unclear what common sense knowledge could be utilized to resolve ambiguity, the disambiguation is not based on general common sense knowledge (as was the case in section §5.1) but instead on asking the participant in the discourse about the meaning of the ambiguous word. Note that although this "asking the participant" discourse strategy is simpler, it relies on a stronger logical machinery which has the ability to represent a sequence of utterances as a discourse. By asking the speaker to clarify themselves we can prove that the GE bid on the FX22 engine is \$2M including spare parts. Note that we will have to state the above in the kb context because the discourse contexts are not expressive enough to distinguish between the price including spares and the price excluding spares (which in fact was the source of ambiguity).

Proposition (kb): $\mathtt{ist}(c_{\mathrm{kb}}, \textit{price-including-spares}(\text{FX22-engine}, \$2\text{M}))$

Proof (kb): By reasoning similar to the first question, we can conclude

(33) $\mathtt{ist}(c6, \textit{price}(x, y)) \leftrightarrow \mathtt{ist}(c_{\mathrm{kb}}, \textit{price-including-spares}(x, y))$

From the frame axioms we get

(34) $$\mathtt{ist}(c6, \textit{price}(\text{FX22-engine}, \$2M))$$

similarly to the frame derivation in the second question. Now the proposition follows from the above formulas. $\square_{\mathbf{kb}}$

7 Conclusions and Future Work

Our original motivation for developing the formal theory of context was to address the problem of generality in AI, as introduced in McCarthy 1987. McCarthy's concern with the existing AI systems has been that they can reason only about some particular, predetermined task. When faced with slightly different circumstances than the ones intended, they need to be completely rewritten to incorporate a new description of the world. In other words, AI systems lack generality. As was argued in McCarthy 1993 and Guha 1991, a logic of context enables a reasoning system to coherently use theories which were not originally intended to be used together.

In this paper, by identifying and focusing on the contexts associated with a particular state of discourse, we have shown how the formal theory of context can be used to represent lexical ambiguity, represent context change in simple discourse segments, and how a general common sense knowledge base can be used to resolve ambiguity. We are currently investigating whether and how other types of ambiguity (see, for example, Hirst 1987) can be handled using the formal theory of context.

Linguists, philosophical logicians, and natural language researchers have already studied similar notions of context. However, in their work context is often represented implicitly, and not treated as a formal object. Treating context as a formal object enables us to represent and reason about contextual phenomena in the framework of logic. In most cases where linguists and philosophers do have explicit theories of context (like DRT and typed lambda calculus, which we discussed in §5.3), these theories lie embedded in the analysis of specific linguistic constructions, so that identifying the relations to our formal theory of context is itself a research challenge. We are currently investigating these relations.

7.1 Acknowledgements

The author thanks Vanja Buvač, Tom Costello, Ian A. Mason, John McCarthy, Johan van Benthem, Kees van Deemter, and the three anonymous reviewers for their valuable comments. This research is supported in part by the Advanced Research Projects Agency, ARPA Order 8607, monitored by NASA Ames Research Center under grant NAG 2-581, by NASA Ames Research Center under grant NCC 2-537, NSF grants CCR-8915663, and IRI-8904611 and Darpa contract NAG2-703.

References

Ahn, R., and H. Kolb. 1990. Discourse representation meets constructive mathematics. In *Papers from the second symposium on logic and language*, ed. L. Kálmán and L. Pólós. Akadémia Kiadó, Budapest.

Barendregt, Henk. 1992. Lambda Calculi with Types. In *Handbook of Logic in Computer Science*, ed. Abramsky, Gabbay, and Maibaum. Oxford University Press.

Buvač, Saša, Vanja Buvač, and Ian A. Mason. 1995. Metamathematics of Contexts. *Fundamenta Informaticae* 23(3). To appear.

De Bruijn, N.G. 1991. Telescopic mappings in typed lambda calculus. *Information and Computation* 91:189–204.

Grosz, Barbara J., and Candace L. Sidner. 1986. Attention, Intention, and the Structure of Discourse. *Computational Linguistics* 12:175–204.

Guha, R. V. 1991. *Contexts: A Formalization and Some Applications.* Doctoral dissertation, Stanford University. Also published as technical report STAN-CS-91-1399-Thesis, and MCC Technical Report Number ACT-CYC-423-91.

Guha, Ramanathan V., and Douglas B. Lenat. 1990. Cyc: A Midterm Report. *AI Magazine* 11(3):32–59.

Hirst, Graeme. 1987. *Semantic Interpretation and the Resolution of Ambiguity.* Cambridge University Press.

Kameyama, Megumi. 1994. Indefeasible semantics and defeasible pragmatics. In *Quantifiers, Deduction, and Context*, ed. Makoto Kanazawa, Christopher Pinon, and Henriette de Swart. CSLI, Stanford, CA. Also SRI International Technical Note 544 and CWI Report CS-R9441,1994.

Kamp, Hans. 1981. A Theory of Truth and Discourse Representation. In *Formal Methods in the Study of Language*, ed. J. Groenendijk, T. Janssen, and M. Stokhof. Amsterdam: Mathematical Centre Tracts 135.

McCarthy, John. 1987. Generality in Artificial Intelligence. *Comm. of ACM* 30(12):1030–1035. Also in *ACM Turing Award Lectures, The First Twenty Years*, ACM Press, 1987; and reprinted in McCarthy 1990b.

McCarthy, John. 1990a. An Example for Natural Language Understanding and the AI Problems it Raises. In *Formalizing Common Sense: Papers by John McCarthy.* 355 Chestnut Street, Norwood, NJ 07648: Ablex Publishing Corporation.

McCarthy, John. 1990b. *Formalizing Common Sense: Papers by John McCarthy.* 355 Chestnut Street, Norwood, NJ 07648: Ablex Publishing Corporation.

McCarthy, John. 1993. Notes on Formalizing Context. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence.*

McCarthy, John, and Saša Buvač. 1994. Formalizing Context (Expanded Notes). Technical Note STAN-CS-TN-94-13. Stanford University.

This paper and other papers on context are available to WWW browsers at `http://sail.stanford.edu/buvac` and are also available by anonymous FTP from `sail.stanford.edu`.