DESIGNER DEFINITES IN LOGICAL FORM

Mary P. Harper*
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907

Abstract

In this paper, we represent singular definite noun phrases as functions in logical form. This representation is designed to model the behaviors of both anaphoric and non-anaphoric, distributive definites. It is also designed to obey the computational constraints suggested in Harper [Har88]. Our initial representation of a definite places an upper bound on its behavior given its structure and location in a sentence. Later, when ambiguity is resolved, the precise behavior of the definite is pinpointed.

1 Introduction

A goal of natural language research is to provide a computer model capable of understanding English sentences. One approach to constructing this model requires the generation of an unambiguous internal representation for each sentence before attempting to represent subsequent sentences. Natural language systems that attempt to guess the intended meaning of a sentence without considering subsequent sentences usually make no provision for recovery from incorrect guesses since that would require storing information about the ambiguity of the sentence. Hence, this approach may require the processing of several sentences before enough information is available to determine the intended meaning of the sentence being represented. However, in order to make the inferences necessary to resolve some ambiguities, some internal representation is needed for both the current sentence as well as subsequent sentences. A more powerful approach is to leave the ambiguity unresolved in an intermediate representation until the necessary information has been processed. We adopt this second approach, which advocates mapping parsed sentences into an intermediate level of representation called *logical form*

[SP84; All87; Har88]. Logical form partially specifies the meaning of a sentence based on syntactic and sentence-level information, without considering the effect of pragmatics and context. Later, as more information becomes available, the representation of the sentence is incrementally updated until all ambiguities have been resolved.

In the literature, two sources of ambiguity have been handled using logical form, quantifier scoping (see [SP84; All87]) and pronoun resolution (see [Har88; Har90]). In this paper, we will discuss the use of logical form for handling the ambiguities in the meanings of singular definite noun phrases. But first, it will be useful to briefly review the logical form for pronouns.

2 Pronouns in Logical Form

Pronouns are a source of underspecification in a sentence which can be handled in logical form. The antecedent of a pronoun cannot be immediately determined when the sentence containing it is parsed. Contextual and syntactic constraints combine to allow a listener/reader to decide on the antecedent for a certain pronoun. In Harper [Har88; Har90], we devised a logical form representation for pronouns. This representation divides the process of determining the meaning of a pronoun into two phases. First, the representation for the pronoun is determined using only syntactic and sentence-level information. Then, once the antecedent is determined, a feat which often requires pragmatic and contextual information available in subsequent sentences, we provide a way to update our logical form to indicate this information.

Our logical form representation for pronouns was designed with two goals in mind. First, we required our representation to be compatible with the goal of devising a computational model of language comprehension. In fact, we defined three constraints for using logical form in a computational framework (from [Har88] and [Har90]).

1. **Compactness Constraint:** Logical form should compactly represent ambiguity.

2. **Modularity Constraint:** Logical form should be initially computable from syntax

and local (sentence-level) semantics. In particular, logical form should not be dependent on pragmatics, which requires inference and hence, internal representation.

3. **Formal Consistency Constraint:** Further processing of logical form should only disambiguate or further specify logical form. Logical form has a meaning. Any further processing must respect that meaning.

First, the compactness constraint captures the spirit of logical form as presented by Allen [All87]. Second, if the modularity constraint is violated, the value of computing logical form is lost. Finally, the formal consistency constraint keeps us honest. Initially, logical form provides a composite representation for a sentence. However, as more information becomes available, then the meaning of the sentence will be incrementally updated until all ambiguity is resolved. We cannot modify logical form in any way that contradicts its original meaning.

The second goal of our approach was to accurately model the linguistic behavior of pronouns while obeying our logical form constraints. Since pronouns have a range of behaviors between variables on the one hand and constants on the other, the initial logical form for a pronoun must be compatible with both extremes (to model the range of pronoun behaviors and to be consistent with the compactness and formal consistency constraints). Hence, we provided a composite representation for a pronoun, one compatible with any possible antecedent it can have given its position in a sentence.

Pronouns in a sentence are represented as part of the process of providing logical form for that sentence. We enumerate the important features of a sentence's representation.

1. A sentence is represented as a predicate-argument structure, with subjects lambda abstracted to handle verb phrase ellipsis. Lambda operators are necessary for handling examples of verb phrase ellipsis. The second sentence in Example 1 is a sentence with verb phrase ellipsis (also called an *elided sentence*).

Example 1

```
Trigger Sentence: Fredᵢ loves hisᵢ wife.
Elided Sentence:  Georgeⱼ does too.
Meanings:
a. George loves Fred's wife.
b. George loves George's wife.
```

Assuming that the meaning of the elided verb phrase is inherited from the representation of the trigger sentence's verb phrase, then the the pronoun *his* in the trigger verb phrase must be able to refer indirectly to the subject *Fred* in

order for the sloppy reading of the elided sentence (i.e., *George loves George's wife*) to be available. All sentences are potentially trigger sentences; hence, we lambda abstract the syntactic subjects of all sentences (following Webber [Web78] and Sag [Sag76]).

2. The logical roles of all noun phrases in a sentence are identified by position in logical form (logical subject first, logical object second, logical indirect object third, etc.).

3. We represent universal noun phrases as universally quantified (and restricted) variables and indefinite noun phrases as existentially quantified (and restricted) variables (following Webber [Web78]).

4. Quantifier scope ambiguity is handled in the same way as in Allen [All87]. Initially, we place quantifiers in the predicate-argument structure (except for subjects). Later, when information becomes available for making scoping decisions, quantifier scoping is indicated (discussed in Harper [Har90]).

A composite representation for a pronoun is provided once the parse tree for the sentence containing it is available. When the parse tree is provided, we can determine all of the quantified noun phrases that are possible antecedents for a pronoun in the sentence (see Reinhart [Rei83]). Hence, we represent a pronoun initially as a function of all of the variables associated with noun phrases that are possible antecedents for or distribute over possible antecedents for the pronoun. To handle verb phrase ellipsis, the argument list must also include the lambda variables corresponding to syntactic subjects. A pronoun is represented as a uniquely-named function of all lambda variables (associated with subjects) which have scope over it in logical form, any non-subject quantified variables corresponding to noun phrases that c-command the pronoun (following Reinhart [Rei83]), and any quantified noun phrase not embedded in a relative clause but contained in a noun phrase that c-commands the pronoun. The lambda variable of a quantified subject subsumes the subject's quantified variable because the lambda operator abstracts the quantified variable. Our logical form representation for pronouns summarizes all of the operators that can directly affect their final meanings. Hence, the representation is useful for limiting the possible antecedents of a pronoun. For example, a pronoun function can take a universal noun phrase as its antecedent if and only if the universal variable (or the variable corresponding to the lambda operator that abstracts the universal variable) is included in the function's argument list.

Consider a simple example to demonstrate the initial representation of the following sentence.

Example 2

```
Every teacher gave every student his paper.
∀x: (teacher x)
  x, λ(y)(give y (paper-of (his₁ y z))
           [∀z: (student z) z])
```

The syntactic subject of the sentence is universally quantified, and the restriction on the quantifier is indicated after the colon[1]. The syntactic subject of the sentence is abstracted from the predicate-argument structure representing the sentence. Hence, the verb phrase, represented as a lambda function, is separable from the subject. The subject's position is maintained in the lambda function by the lambda variable. Notice that the definite noun phrase *his paper* is represented here as a function of the pronoun. Shortly, we will provide a more general representation for definite noun phrases. Notice that the pronoun *his* is represented as a function of subject's lambda variable plus the universal variable corresponding to *every student*. This list of arguments corresponds to the operators for noun phrases that can be antecedents for the pronoun given the syntactic constraints or can distribute over possible definite antecedents. Notice that the subject's lambda variable subsumes the subject's universal variable. The reader should note that quantifier scoping is not indicated in our initial logical form (following Allen [All87]).

The representation for the pronoun in 2 is a composite representation, that is it indicates all of the operators that can affect its final meaning. In fact, before the final meaning of the sentence can be given, the antecedent for the pronoun must be determined and made explicit in our logical form. Though the process of determining antecedents for pronouns is beyond the scope of this paper, when a pronoun's antecedent is known (requiring additional pragmatic information), the logical form containing it must be updated in a way compatible with its initial representation (because of the formal consistency constraint). Suppose that we decide that the antecedent for *his* in example 2 is *every student*, then the logical form is be modified as shown in 3.

---

[1]The colon following the quantifier is syntactic sugar which expands the restriction differently depending on the type of quantifier. If a sentence is represented as $∃x$: (R x) (P x), then the meaning is $∃x$ (and (R x) (P x)). If a sentence is represented as $∀x$: (R x) (P x), then it is expanded as $∀x$ (if (R x) (P x)).

Example 3

```
Every teacherᵢ gave every studentⱼ hisⱼ paper.
∀x: (teacher x)
  x, λ(y)(and (give y (paper-of (his₁ y z))
               [∀z: (student z) z])
          (= (his₁ y z) z))
```

This update is compatible with the pronoun's initial representation. We are indicating that the function (his₁ y z) is really the identity function on z. In Harper [Har88], we fully specify how logical form is updated when a pronoun's antecedent has been determined.

3 Definites: Behaviors to Cover

In the rest of this paper, we develop our logical form representation for singular definite noun phrases. As for pronouns, we wish to obey our computational constraints while providing a good model of definite behavior. Consider the behaviors of definites we wish to cover.

Like pronouns, definite noun phrases can be anaphoric. Anaphoric definites can either depend on linguistic antecedents (in either the same or previous sentences) or can denote salient individuals in the environment of the speaker/hearer (also called deictic use). Because of our logical form constraints, in particular because of the compactness and formal consistency constraints, the initial representation for a definite noun phrase must be compatible with the representations of its possible antecedents. Definite noun phrases can have intrasentential antecedents as in example 4.

Example 4

```
Every boyᵢ saw (hisᵢ dog)ⱼ before the beastⱼ
saw himᵢ.
```

In this case, the definite noun phrase acts like a universally quantified variable (adopting the behavior of its antecedent in much the same way as a pronoun).

Definites, unlike pronouns, can also have a complex syntactic structure. Pronouns and other noun phrases can be attached to a definite noun phrase in different ways. First, consider the effect embedded pronouns have on definite noun phrases. While simple definites (which are not intrasentential anaphors) seem to act like constants when they occur in a sentence with a universal noun phrase (e.g., 5a), definite noun phrases with embedded pronouns often cannot be described as constants (e.g., 5b).

Example 5

```
a. Every boy loves the woman.
b. Every boy loves his mother.
```

The meaning of *his mother* depends on how the pronoun is resolved. If the antecedent for *his* is found in another sentence, then *his mother* could be represented as a constant. In contrast, if *every boy* is the antecedent for *his*, then the universal quantifier corresponding to *every boy* distributes over *his mother*. When a quantifier distributes over a definite, the definite changes what it denotes based on the values assigned to the quantified variable.

Embedded quantified noun phrases can also distribute over a definite noun phrase, preventing it from acting like a constant. For example, the universal possessive noun phrase distributes over the definite in the following sentence. The definite in this case cannot be described as a constant.

Example 6

`George loves every man's wife.`

However, not all embedded quantified noun phrases can distribute over a definite. When quantified noun phrases are embedded in relative clauses attached to a definite noun phrase, they cannot distribute over that noun phrase. This constraint (related to the complex noun phrase constraint, first noted by [Ros67]) prohibits quantifiers from moving out of a relative clause attached to a noun phrase. For example:

Example 7

`George saw the mother who cares for every boy.`

In this case, *the mother who cares for every boy* denotes one specific mother. In such cases, the universal cannot distribute over the definite it is attached to or have scope over other quantified noun phrases outside of the relative clause.

Thus, the meaning of a definite noun phrase is affected by its structure, whether it contains pronouns, and whether or not it is used anaphorically. If used anaphorically, it should behave in a way consistent with its antecedent, just like a pronoun. If it contains pronouns, then its meaning should depend on the antecedents chosen for those pronouns. If it contains embedded quantified noun phrases (not subject to the relative clause island constraint), then those embedded noun phrases may distribute over the definite.

In the remainder of this paper, we introduce our logical form representation for definites. We discuss the initial representation of definites, which must be able to encompass all of the above definite behaviors. We also describe the ways this logical form is updated once ambiguity is resolved.

4 Our Representation of Definite Noun Phrases

In this section, we develop a representation for definites in logical form. The logical form representation for a definite noun phrase presents a challenge to our approach. To be consistent with the modularity constraint, we must provide an initial representation for a definite noun phrase that can be generated before we know the antecedents for any embedded pronouns or before we know the definite's antecedent (if it is anaphoric). To obey the compactness and formal consistency constraints, we must initially represent a definite so it is consistent with all the ways it can possibly act. As more information becomes available about the meaning of the definite noun phrase, we must be able to update logical form in a way compatible with its initial representation. Our logical form for a definite must be a composite representation compatible with its possible behaviors. We cannot provide different initial representations for a definite depending on use, otherwise we violate the compactness constraint. Additionally, unless our initial representation is compatible with all possible behaviors, we could violate the formal consistency constraint when we update logical form.

We represent a definite as a named function of all of the variables associated with operators that can affect its meaning. This representation satisfies our constraints by combining the advantages of definite descriptions (discussed in Harper [Har90]) with the functional notation we introduced to represent pronouns. Each definite function is defined by a unique name (i.e., *def* with a unique integer appended to it), a list of arguments, and a restriction. The restriction of a definite function is derived from the words following the determiner. The argument list of the function consists of the variables associated with lambda operators that have scope over its position, any variables associated with non-subject quantified noun phrases that could bind a pronoun in that position, and any quantified variables associated with embedded quantified noun phrases that are not embedded in a relative clause attached to a noun phrase[2]. Because a definite function has a unique name, we can differentiate two occurrences of the same definite noun phrase, in contrast to definite descriptions [Rus71] (for more information on the shortcomings of definite descriptions and definite quantifiers, see [Har90; Hin85]).

---

[2]We should also add that a sententially attached PP with a quantified object can quantify over a definite as well (as in, *In every car, the driver turned the steering wheel.* This sentence is tricky because we seem to be attaching the PP to both of the NPs while leaving the quantifier to distribute over both definites).

Consider the initial representation of a sentence containing a definite noun phrase before the antecedent of an embedded pronoun is known:

Example 8

```
Every man showed every boy his picture.
∀x: (man x)
  x, λ(y)(show y
                ((def₁ y z) |
                    (and (picture (def₁ y z))
                         (possess (his₂ y z)
                                  (def₁ y z))))
                 [∀z: (boy z) z])
```

The representation of this sentence is very similar to example 2 except for the representation of the definite noun phrase. Notice that *his picture* is represented as a function called def_1. The restriction of the function is the conjunction of statements following the vertical bar. The vertical bar is syntactic sugar and should be expanded like the colon in an existential's restriction (but not until the definite's final meaning is determined). The argument list of the function consists of the variables y and z[3]. Notice that the pronoun *his* is also represented as a function of y and z. Anything that can affect the pronoun *his picture* will also affect the meaning of the definite noun phrase.

Because a definite function is a composite representation for all possible meanings of a definite noun phrase, we must restrict the function in certain ways before a final interpretation for the sentence is available (or before deriving the meaning of an elided sentence from a trigger verb phrase containing a definite function, as discussed in [Har90]). The initial representation of a definite places an upper and lower bound on the definite's behavior. The lower bound is a constant, while the upper bound is the initial representation. These bounds must be tightened to settle on a final interpretation for the definite. We provide two methods to pinpoint a definite function. If the definite is used anaphorically, we equate the definite function with some value consistent with its antecedent. Otherwise, we apply a constraint that limits the argument list of the function to include only necessary variables.

If a definite is used anaphorically, it can be equated with some value depending on its antecedent (just like pronoun functions in [Har88]). For example, if the antecedent of a definite noun phrase occurs in another sentence, we would equate the definite function with a discourse entity. Antecedents for definite noun phrases can also occur

within the same sentence. An intrasentential reference to an antecedent requires the definite function to have an argument list compatible with the representation of the antecedent[4]. Consider the initial representation of a sentence containing a potentially anaphoric definite shown in 9.

Example 9

```
Every man told his mother's psychiatrist about
the old lady's diary.
∀x: (man x)
  x, λ(y)(tell
              y
              ((def₁ y) |
                  (and (psychiatrist (def₁ y))
                       (possess
                          ((def₂ y) |
                              (and (mother (def₂ y))
                                   (possess
                                      (his₃ y)
                                      (def₂ y))))
                           (def₁ y))))
           (about
              ((def₄ y) |
                  (and (diary (def₄ y))
                       (possess
                          ((def₅ y) |
                              (old-lady (def₅ y)))
                           (def₄ y)))))))
```

Suppose the antecedent for *his* is *every man* and the antecedent for *the old lady* is *his mother*. Then we can augment the logical form, as shown in 10.

---

[3] As in the representation of pronouns, we omit the variable x from the argument list because the lambda operator for y abstracts x, so y is the more general argument.

[4] It is unusual for a definite to have an antecedent corresponding to one of its arguments unless the variable corresponds to a quantified noun phrase which is not embedded in a relative clause but is embedded in another noun phrase. When the antecedent is represented as a function, its argument list must be a subset of (or it must be possible to limit it to be a subset of) the arguments of the anaphoric definite for the equality to be asserted.

Example 10

```
Every man_j told (his_j mother's)_i psychiatrist
about the old lady's_i diary.
∀x: (man x)
  x, λ(y)(tell
          y
          ((def_1 y) |
              (and (psychiatrist (def_1 y))
                   (possess
                      ((def_2 y) |
                          (and (mother (def_2 y))
                               (possess (his_3 y)
                                        (def_2 y))
                               (or (= (his_3 y) y)
                                   (= (his_3 y) x))))
                      (def_1 y))))
          (about
             ((def_4 y) |
                 (and (diary (def_4 y))
                      (possess
                         ((def_5 y) |
                             (old-lady (def_5 y)))
                         (def_4 y))
                      (= (def_5 y) (def_2 y)))))))
```

This example would be very difficult for an approach that uses either definite descriptions or definite quantifiers. Either approach would represent *the old lady* in a way equivalent to replacing the representation by a constant, because of uniqueness. Hence, any update of those representations to indicate the anaphora would violate formal consistency. Our approach, however, can easily handle the example.

The other way to pinpoint a definite function applies once antecedents for embedded pronouns are known and once we know whether quantifiers corresponding to embedded quantified noun phrases (not embedded in relative clauses attached to noun phrases) should distribute over the definite. Consider the initial representation of the sentence in 8. The definite function def$_1$ is a function of all of the variables that can potentially cause it to change. However, once we know the antecedent for its embedded pronoun, the argument list of the function should be limited. To limit the argument list, we make use of the insights gained from definite descriptions. Because of the uniqueness assumption, any definite description that does not contain variables bound by outside quantifiers acts like a constant. On the other hand, if a pronoun embedded in a definite description adopts the behavior of a universally quantified variable, then the definite description will change what it denotes depending on the instantiation of that variable. Hence, we conclude that a definite function should only change as a function of those variables bound by operators outside of its restriction (ignoring its own argument list).

Once antecedent and embedded quantifier information is available, we can limit the argument list to precisely those arguments that are bound by operators outside of the restriction. If a pronoun function in the restriction of the definite function is equated with a variable bound outside its restriction or with another function which must be a function of a certain variable (based on its own restriction), then the argument must be retained. Additionally, other arguments that are free in the restriction must be retained (these correspond to embedded quantified noun phrases whose quantifiers are moved out of the restriction). Once we know the necessary arguments, we replace the original function by a new function over those arguments. By using this argument reduction constraint, we limit the initial composite representation of a definite noun phrase to its final meaning (given pronoun and quantifier information).

Consider how we would limit the function (def$_1$ y z) from example 8 following pronoun resolution. If we decide that the antecedent of *his* is *every boy*, then we would update the logical form, as shown in 11.

Example 11

```
Every man showed every boy_i his_i picture.
∀x: (man x)
  x, λ(y)(show y
              ((def_1 y z) |
                  (and (picture (def_1 y z))
                       (possess (his_2 y z)
                                (def_1 y z))
                       (= (his_2 y z) z)))
              [∀z: (boy z) z])
```

By using our argument reduction constraint, we can replace the function (def$_1$ y z) by a function of z (since (his$_2$ y z) is replaced with the variable z), as shown in 12.

Example 12

```
Every man showed every boy_i his_i picture.
∀x: (man x)
  x, λ(y)(and (show
               y
               ((def_1 y z) |
                   (and (picture (def_1 y z))
                        (possess (his_2 y z)
                                 (def_1 y z))
                        (= (his_2 y z) z)))
               [∀z: (boy z) z])
          (= (def_1 y z) (def_3 z)))
```

Equality here is equivalent to replacing the first function with the second value. Because of this fact and because of the meaning of the vertical bar in the restriction of the function, this representation can be simplified as shown in 13.

67

Example 13

Every man showed every boy; his; picture.
```
∀x: (man x)
  x, λ(y)(and (show y
                   (def₃ z)
                   [∀z: (boy z) z])
             (picture (def₃ z))
             (possess z (def₃ z)))
```

To handle the readings where *his* is anaphorically dependent on other noun phrases, our approach would be similar.

Our representation of pronouns has several strengths. First, the representation provides useful information to a semantic routine concerning possible intrasentential antecedents for the definite. Argument lists limit what can be the antecedent along with other factors like number and gender agreement and antecedent limitations particular to definites. To demonstrate a strength of this approach, consider the initial representation of the following sentence:

Example 14

Fred told the teacher who discusses every student with his mother to record her response.
```
((def₁) | (name (def₁) Fred)),
  λ(x)(tell
        x
        ((def₂ x) |
          (and (inst (def₂ x) teacher)
               (def₂ x),
               λ(y)(discuss
                       y
                       [∀(z):
                         (inst z student) z]
                       (with
                         ((def₃ x y z) |
                           (and
                             (inst
                               (def₃ x y z)
                               mother)
                             (possess
                               (his₄ x y z)
                               (def₃ x y z)))))))))
        [(def₂ x),
          λ(w)(record
                 w
                 ((def₅ x w) |
                   (and (inst (def₅ x w)
                              response)
                        (possess
                          (her₆ x w)
                          (def₅ x w)))))])
```

Here the meaning of *her response* depends on the antecedent for *her*. What then are legal antecedents for *her* in this sentence? Certainly, *the teacher* is a fine candidate, but what about *his mother*. We cannot tell immediately whether *his mother* can be the

antecedent for *her*. If the antecedent for *his* is *every student*, then his mother cannot be the antecedent for *her*. This accessibility problem results because the universal in the relative clause (i.e., *every student*) cannot have scope over *her response*, hence, *his mother* is not a good antecedent for *her*[5]. Notice that (her₆ x w) is not immediately compatible with the representation for *his mother* (i.e., (def₃ x y z)). Before we can assert that *his mother* is the antecedent for *her* we must pinpoint the meaning of that noun phrase, that is, we must determine the antecedent for *his*. Then depending on our choice, the final meaning of *his mother* may or may not be accessible to the pronoun. Hence, we can explain why some definites in relative clauses are accessible to pronouns in the matrix sentence and others are not. C-command does not accurately predict when definites are accessible as antecedents for anaphoric expressions. This is not surprising, given the fact that the final meaning of a definite determines its accessibility, and determining this meaning may require resolving pronouns and scoping ambiguities.

In this paper, we have introduced a composite representation for definite noun phrases with two ways to update their meaning as more information becomes available. This approach is consistent with the three computational constraints discussed in section 2, and also provides a good model of definite behavior. We refer the reader to Harper [Har90] for discussion of a wider variety of examples. In particular, we discuss examples of verb phrase ellipsis, Bach-Peters sentences, and definite donkey sentences [Gea62]. Our approach has been implemented and tested on a wide variety of examples. The logical form for pronouns and definites is provided as soon as a parse tree for the sentence is available. Then, the logical form for the sentence is incrementally updated until all ambiguities have been resolved. Logical form is very useful in the search for pronoun and definite antecedents. For more on the implementation see [Har90].

One shortcoming of our approach is our inability to provide a single logical form for a sentence with structural ambiguity. One possible solution to this problem (which we are currently investigating) is to store partial logical forms in a parse forest. As more information is processed this intermediate representation will be incrementally updated until the parse forest is reduced to a single tree containing

---

[5]Strictly speaking, universal noun phrases cannot bind across sentences. However, speakers sometimes allow a universal to be the antecedent for a singular pronoun outside of its scope. Such pronouns are not usually understood as giving a bound variable reading. See Webber [Web78] for a discussion of this issue. A similar treatment can apply to definites which change as a function of a universal.

one logical form.

5 Past Approaches

Our work has benefited from the insights gained from other approaches to definite noun phrases in the literature. We considered both definite descriptions introduced by Russell [Rus05] and definite quantifiers (used by many including [Web83]) for representing definite noun phrases. Neither representation allows us to handle intrasentential anaphoric definites while obeying our computational constraints. However, the in-place definite description is excellent for modeling definite subjects in verb phrase ellipsis and for capturing the behaviors of distributive definite noun phrases. On the other hand, a definite quantifier is not a good representation for a definite subject in verb phrase ellipsis (the strict meaning of *The cat wants its toy. The dog does too* cannot be provided because quantifiers do not have scope across sentences). In fact, to make the definite quantifier a feasible representation, we would have to make the binding properties of a definite quantifier different than the binding properties of a universal. Hornstein [Hor84] suggests that definite quantifiers have different binding properties than universals. His approach fails to consider how the process of pinpointing the meaning of a definite affects its ability to bind a pronoun. For more discussion of the strengths and weaknesses of these approaches, see Harper [Har90].

Other approaches to handling definites include the work of [Hei82; Kam81; Rob87; Kle87; PP88]. Each approach differs from ours both in scope and emphasis. We build an intermediate meaning for a sentence using only the constraints dictated by the syntax and local semantics and incrementally update it as we process contextual information. The work of Pollack and Periera [PP88] also attempts to gradually build up a final interpretation of a sentence using their semantic and pragmatic discharge interpretation rules. However, our representation of a definite noun phrase locally stores information about those quantifiers in the sentence that can potentially quantify over it, while Pollack and Periera's representation does not. The approaches of [Hei82; Kam81; Rob87; Kle87] require a large amount of contextual information before the representation of a sentence can be given (leading to a violation of our constraints).

References

[All87] James Allen. *Natural Language Understanding*. The Benjamin/Cummings Publishing Company, Menlo Park, CA, 1987.

[Gea62] Peter T. Geach. *Reference and Generality*. Cornell University Press, Ithaca, 1962.

[Har88] Mary P. Harper. Representing pronouns in logical form : Computational constraints and linguistic evidence. In *The Proceedings of the 7th National Meeting of AAAI*, 1988.

[Har90] Mary P. Harper. *The representation of noun phrases in logical form*. PhD thesis, Brown University, 1990.

[Hei82] Irene Heim. *The Semantics of Definite and Indefinite Noun Phrases*. PhD thesis, University of Massachusetts, 1982.

[Hin85] Jaakko Hintikka. *Anaphora and Definite Descriptions: Two applications of Game-Theoretical semantics*. D. Reidel Publishing Company, Boston, 1985.

[Hor84] Norbert Hornstein. *Logic as Grammar: An Approach to Meaning in Natural Language*. MIT Press, Cambridge, MA, 1984.

[Kam81] Hans Kamp. A theory of truth and semantic representation. In Jeroen Groenendijk, Theo Janssen, and Martin Stokhof, editors, *Formal Methods in the Study of Language*, volume 1. Mathematische Centrum, Amsterdam, 1981.

[Kle87] Ewan Klein. VP ellipsis in DR theory. In J. Groenendijk, D. de Jongh, and M. Stokhof, editors, *Studies in Discourse Representation and the Theory of Generalized Quantifiers*. Foris, Dordrecht, 1987.

[PP88] Martha E. Pollack and Fernando C.N. Pereira. An integrated framework for semantic and pragmatic interpretation. In *The Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, 1988.

[Rei83] Tanya Reinhart. *Anaphora and Semantic Interpretation*. Croom Helm, London, 1983.

[Rob87] Craige Roberts. *Modal Subordination, Anaphora, and Distributivity*. PhD thesis, University of Massachusetts, 1987.

[Ros67] John R. Ross. *Constraints on Variables in Syntax*. PhD thesis, MIT, 1967.

[Rus05] Bertrand Russell. On denoting. *Mind*, 14:479–493, 1905.

[Rus71] Bertrand Russell. Reference. In J. F. Rosenberg and C. Travis, editors, *Readings in the Philosophy of Language*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.

[Sag76] Ivan A. Sag. *Deletion and Logical Form*. PhD thesis, MIT, 1976.

[SP84] L. K. Schubert and F. J. Pelletier. From English to Logic : Context-free computation of 'conventional' logical translations. *American Journal of Computational Linguistics*, 10:165–176, 1984.

[Web78] Bonnie L. Webber. *A Formal Approach to Discourse Anaphora*. PhD thesis, Harvard, 1978.

[Web83] Bonnie L. Webber. So what can we talk about now? In M. Brady and R. Berwick, editors, *Computational Models of Discourse*. MIT Press, Cambridge MA, 1983.