UTPAL LAHIRI

GENERALIZED QUANTIFIERS IN LOGIC AND NATURAL LANGUAGE

ABSTRACT: This survey article is intended as a very basic introduction to the theory of Generalized Quantifiers (GQs) and Determiners in Logic and natural language, mainly addressed to an audience consisting of those with some familiarity with linguistic theory and some rudimentary (at best) knowledge of first order predicate logic. The theory of GQs is first motivated by a consideration of examples from quantified statements in natural language, and a compositional semantics is presented for natural language sentences that contain GQs. Various mathematical properties of GQs and (semantic) Determiners, like Conservativity, Symmetry, Monotonicity properties, are then introduced and their relevance to linguistic phenomena are discussed. In the last segment, the concept of isomorphism invariance is introduced and discussed, along with some very basic mathematical properties of logical determiners that derive from them. Some brief comments on non-logical determiners (mainly possessives) follow and round up the discussion.

KEYWORDS: Generalized Quantifiers, Semantic Determiners, Logical Form, Conservativity, Symmetry, Extensionality, Monotonicity, Downward Entailment, Isomorphism Invariance, Possessive Descriptions.

0. INTRODUCTION

Modern, formal logic was developed in the late 19th and early 20th centuries in the work of philosophers and mathematicians (most importantly Frege, Russell and Whitehead and Peano, among others), though it has antecedents in the western philosophical tradition (starting with Aristotle, going through Boole and many others) as well as in other philosophical traditions, most prominently in the Arabic, Indian as well as Chinese, particularly in its Indian-influenced Buddhist traditions (for a survey of the development of logic in the west, see Kneale and Kneale (1971)). The best known and most popular logical system that is the result of such investigations is first order predicate logic with identity. In this article, we examine a fuller range of quantified expressions in

The EFL Journal 8:1 January 2017.
©2017 The English and Foreign Languages University
English, which have counterparts in many languages of the world. We note that first-order logic has some inherent limitations as a theory of quantification in natural language in its full generality. The theory of Generalized Quantifiers was first proposed in logic as a way to capture the behavior of a broader class of quantifiers than simply the universal and existential quantifiers. It is based on an insight of the philosopher Frege that the “quantifiers” can be viewed as relations between two predicates, and this perspective has far-reaching implications for the treatment of quantification in natural language. The earliest formulations in logic of the theory of generalized quantifiers can be found in the work of the Polish mathematician Mostowski (Mostowski 1957), and further developed for the case of restricted quantification (closer to what is found in natural language) in the work of the Swedish logician Paul Lindström (Lindström 1966). In this article, we go through some of the reasons for why we need this extension, both from the point of view of expressiveness as well as from considerations of compositionality, in the context of an adequate theory of the syntax-semantic mapping. Later, we will also look at some universal constraints on the meanings of the determiners that Generalized Quantifiers are composed of, as well as some other properties that define classes of Determiner meanings that have consequences for the grammars of natural languages. A basic familiarity with first-order logic is presupposed for readers of this article.

1. QUANTIFICATION IN FIRST ORDER LOGIC AND IN NATURAL LANGUAGE

It is well known that sentences of English that contain quantifier expressions like every, some, etc. can be translated in first-order logic by formulas that contain quantifier expressions like ∀ or ∃ and symbols for (non-logical) constants, variables, n-place predicate letters. Thus, the sentences (1) and (2) can be translated as (3) and (4), respectively:

(1) Every donkey snores.
(2) Some donkey snores.
(3) ∀x (P(x)→Q(x))
(4) ∃x(P(x)∧Q(x))
(Here “P” and “Q” stand for donkey and snores, respectively.)
There are a couple of problems, however. The syntax of first-order logic departs from that of natural language in significant ways: in the latter, quantified expressions are formed by combining determiner-like expressions like *every* and *some* with other nominal constituents to form phrases that can be analyzed as NPs (Noun Phrases) or DPs (Determiner Phrases), depending on one’s favored syntactic analysis, which then combine with a VP to form sentences. The nominal constituent (“donkey” in examples (1) and (2) above) is called the restriction of the determiner (or determiner-like) expression (“every”, “some” in the above examples), which then combines with a VP to form an expression that denotes a complete thought, i.e., a sentence that can be true or false. In technical terms, quantification in natural language is restricted by the nominal constituent that follows, whereas formulas of first-order logic are unrestricted in the sense that they contain a quantifier symbol with a variable followed by a sentence-like constituent, a “formula”. Moreover, the full class of quantified expressions in natural language is much larger than those that contain every and some: the following sentences of English also are, in some sense, “quantified” expressions:

(5) No donkey snores.
(6) At least two donkeys snore.
(7) At most two donkeys snore.
(8) Many donkeys snore.
(9) Few donkeys snore.
(10) A few donkeys snore.
(11) Between two and twenty five donkeys snore.
(12) Most donkeys snore.

Some of these sentences can be translated in first-order logic. Thus, one can easily see that (5), (6) and (7) receive the following translations, respectively:

(13) \(\neg\exists x(P(x)\land Q(x))\)
(14) \(\exists y\exists x(P(x)\land Q(x)\land P(y)\land Q(y)\land x\neq y)\)
(15) \(\forall x\forall y\forall z((P(x)\land Q(x)\land P(y)\land Q(y)\land P(z)\land Q(z)) \rightarrow (x=y\lor y=z\lor x=z))\)
Seeing that these are equivalent is indeed sometimes somewhat tricky, but you can convince yourself that this is indeed the case. Similarly, the sentence (11) can be decomposed into “at least two and at most twenty five” and one can give a first-order logic translation along the lines of (14) and (15) above. (Exercise: show this). However, there are two problems with this. First, it is obvious that there is no mechanical process by which to translate sentences of English containing quantifiers into first order logic. Thus, the sentence every donkey snores must be translated as (3), with the connective “→”, whereas the sentence some donkey snores must be translated as (4), with the connective “∧”. There is no principle, or algorithm that tells us why it should be one connective rather than the other: it is just that the meanings of the two English sentences are best expressed by the two formulas. The difference clearly has to do with the difference in the meaning of every vs. some, but the first order logic translations must be arrived at by guess work. As one can easily see from the formulas (13)-(15), the relationship between the sentences of English that contain quantifiers and their first-order logic translations only gets more complicated and non-transparent as one moves to “quantifiers” like no, at least two, at most two, and so on.

The second, and more serious, problem is that sentences containing “quantifiers” like most cannot, in principle, be translated into formulas of first-order logic. This can be proved mathematically, but this is something that will not be attempted here (interested readers who are more mathematically sophisticated may consult the paper Barwise and Cooper 1981). While the exact semantics of most is somewhat complex, it can often be taken to mean more than half. Thus the sentence (16) has the same meaning as (17):


It can also be seen easily that (17) is equivalent to (18):

(18) The number of Americans who voted for Obama is greater than the number of Americans who did not vote for Obama.
Barwise and Cooper (1981) show that sentences like (16)-(18) cannot be translated into first-order logic in principle.

2. GENERALIZED QUANTIFIERS AND THE MEANINGS OF NOUN PHRASES

An alternative way of thinking about the semantics of NPs comes from some remarks by the philosopher Frege, who was one of the founders of modern logic. While Frege’s works are often seen as the precursor to the notation of modern logic developed by Alfred North Whitehead and Bertrand Russell, Frege also remarked in passing that quantifier expressions could be seen as second-order relations between what he called “concepts”, which, in our terms, simply means “predicates”. What this means is that a sentence like every donkey snores simply means that the extension of the predicate donkey is included in the extension of the predicate snores, and that a sentence like some donkey snores means that the extension of the predicate donkey overlaps with the extension of the predicate snores, and so on. This can be described as follows in set-theoretic terminology:

(19) Every A is a B is true iff $A \subseteq B$
(20) Some A is a B is true iff $A \cap B \neq \emptyset$

The meanings of the quantifier expressions (every, some and all the rest) can similarly be elucidated in set-theoretic terminology as follows, if we take Frege’s suggestion that these expressions are second-order relations between predicate extensions ($n$ is taken to be an arbitrary natural number in the following examples):

(21) every (A, B) $\leftrightarrow A \subseteq B$
(22) some (A, B) $\leftrightarrow A \cap B \neq \emptyset$
(23) no (A, B) $\leftrightarrow A \cap B = \emptyset$
(24) at least n (A, B) $\leftrightarrow |A \cap B| \geq n$
(25) at most n (A, B) $\leftrightarrow |A \cap B| \leq n$
(26) exactly n (A, B) $\leftrightarrow |A \cap B| = n$
(27) most (A, B) $\leftrightarrow |A \cap B| \geq |A-B|$
(28) the (A, B) $\leftrightarrow$ $A \subseteq B$ and $|A| = 1$
(29) both (A, B) $\leftrightarrow$ $A \subseteq B$ and $|A| = 2$
(30) the n (A, B) $\leftrightarrow$ $A \subseteq B$ and $|A| = n$

The “vague” quantifiers like *many*, *few* and *a few* can similarly be described as follows:

(31) many (A, B) $\leftrightarrow$ $|A \cap B| \geq k$, where $k$ is a contextually determined (large) number.
(32) few (A, B) $\leftrightarrow$ $|A \cap B| \leq k$, where $k$ is a contextually determined (small) number.
(33) a few (A, B) $\leftrightarrow$ $|A \cap B| \geq k$, where $k$ is a contextually determined (small) number.

The right hand sides of the equivalencies given above are determined by the meanings of the quantifiers on the left-hand side, and this provides a perspective on quantifiers that is missing in the notation of first-order logic. It is also more expressive in that it allows one to express a broader class of quantified statements that are present in natural language but cannot be expressed in first-order logic.

A related issue is the question of the compositional interpretation of quantified statements. In customary treatments of *every* and *some* (i.e., simple extensions of the standard semantics of first-order logic to a natural language like English), we use rules of semantic interpretation that are syncategorematic, i.e., the interpretation rules have the following form:

(34) $\langle [\text{every } \beta], S \rangle^{M,g} = 1$ iff ...

Rules of this kind are not strictly compositional in the sense that the semantic values are not assigned to immediate sub-constituents but rather, a rule is provided that assigns a value by looking at a certain complex structure with constituents embedded further down and assigns a value to the entire structure. This is undesirable as we prefer a system with as few semantic rules as possible. This means that we would prefer to have a system where the rule of function application would be able to do much of the job. This would also mean, for example, that we need a way of assigning values to NPs like *every donkey* that the rule system of standard first order logic cannot provide. We now outline a way to get around the problem.
To see the point, consider the NP *every donkey* in the sentence *every donkey snores*. We know that the VP *snores* is a one-place predicate, and so belongs to the semantic type <e, t>. The entire sentence, of course can be true or false, and thus, is of type t. Now, those familiar with basic formal semantics are aware that proper names like *John* are of type e, and so the semantic value of *John* can apply as an argument to the function denoted by *snores*, as the types are the right ones that allow for a correct application of the function application rule. However, we know that NPs like *every donkey* cannot be of type e as they don’t denote things. This means that if we want the rule of function application to apply, *every donkey* must be a function that takes the semantic value of the VP as an argument and yields a truth value. This means that every donkey must be a function from things of type <e, t> to things of type t. In other words, they must be of semantic type <<e, t>, t>.

So what kind of semantic value must it have? So far we have been talking in terms of functions, but we can also think in terms of sets (these are more intuitively clearer than functions). On this way of looking at things, VPs denote sets and sentences like *John snores* are taken to be true if and only if \(|\{John\}| \in |\{snores\}|\). Thinking in terms of set denotations, the semantic values of NPs can now be taken to be sets of sets, as their function denotations are of type <<e, t>, t>. On this view, then,

\[(35) \quad |[\{\text{every donkey}\}, [\text{snores}]]| = 1 \iff |\{\text{snores}\}| \in |\{\text{every donkey}\}|
\]

The set denotation of the predicate *donkey* is a set, and therefore for the equivalence in (35) to hold, it must be that

\[(36) \quad |\{\text{snores}\}| \in |\{\text{every donkey}\}| \iff |\{\text{donkey}\}| \subseteq |\{\text{snores}\}|
\]

This means that

\[(37) \quad |\{\text{every donkey}\}| = \{X \subseteq U \mid |\{\text{donkey}\}| \subseteq X\},\]

where \(U\) is the universe of things.

By exactly parallel reasoning, the set denotation of other quantified NPs can be given as follows:
(38) \[ \{ \text{some donkey} \} = \{ X \subseteq U \mid \text{donkey} \cap X \neq \emptyset \} \]

(39) \[ \{ \text{no donkey} \} = \{ X \subseteq U \mid \text{donkey} \cap X = \emptyset \} \]

(40) \[ \{ \text{at least n donkeys} \} = \{ X \subseteq U \mid \text{donkey} \cap X \geq n \} \]

(41) \[ \{ \text{at most n donkeys} \} = \{ X \subseteq U \mid \text{donkey} \cap X \leq n \} \]

(42) \[ \{ \text{exactly n donkeys} \} = \{ X \subseteq U \mid \text{donkey} \cap X = n \} \]

(43) \[ \{ \text{most donkeys} \} = \{ X \subseteq U \mid \text{donkey} \cap X \geq |\text{donkey} - X| \} \]

The reader can now try to compute the formulas for set denotations for other NPs containing other quantifier expressions.

Before we proceed further, some terminological clarification is in order. So far, we have been using the term “quantifier” to mean expressions like \textit{every} and \textit{some}. This followed the standard practice in the study of logic. However, in later developments, especially in linguistic (formal) semantics, it is customary to use the term “generalized quantifier” to describe the denotations of NPs of the type found in (38)-(43). Denotations of expressions like \textit{every}, \textit{some}, etc. are called “(semantic) determiner” denotations. From now on, I will use this terminology, and use the term “determiner” to mean semantic determiner, unless stated otherwise. Note that semantic determiners used in this sense may or may not be determiners in a syntactic sense. In fact there is much research in current linguistic semantics that further decomposes the meaning of some semantic determiners like “more than n” “less than n”, starting from the observation that these expressions syntactically have a comparative form and this has been shown to be relevant when one analyzes the semantics of the full range of examples that contain these (semantic) determiners (for a survey of some of these issues, see Szabolcsi 2010).

With these considerations in mind, we can also form an idea of what the denotations of determiners must look like. We have established that generalized quantifiers are of semantic type \langle \langle e, t \rangle, t \rangle. Since they are formed by combining a determiner with a 1-place predicate (type \langle e, t \rangle), the determiners themselves must have the semantic type of a function that takes elements of type \langle e, t \rangle as arguments and yields as values elements of type \langle \langle e, t \rangle, t \rangle. In other words, they must be of type \langle \langle e, t \rangle,
The denotations of the determiners would now be functions whose values are given as follows:

\begin{align*}
(44) \quad \text{every}(Y) &= \{ X \subseteq U \mid Y \subseteq X \}, \\
(45) \quad \text{some}(Y) &= \{ X \subseteq U \mid Y \cap X \neq \emptyset \} \\
(46) \quad \text{no}(Y) &= \{ X \subseteq U \mid Y \cap X = \emptyset \} \\
(47) \quad \text{at least n}(Y) &= \{ X \subseteq U \mid \lvert Y \cap X \rvert \geq n \} \\
(48) \quad \text{at most n}(Y) &= \{ X \subseteq U \mid \lvert Y \cap X \rvert \leq n \} \\
(49) \quad \text{exactly n}(Y) &= \{ X \subseteq U \mid \lvert Y \cap X \rvert = n \} \\
(50) \quad \text{most}(Y) &= \{ X \subseteq U \mid \lvert Y \cap X \rvert \geq \lvert Y - X \rvert \}
\end{align*}

One can similarly extend this to a variety of determiners mentioned above in the text not explicitly mentioned in (44)-(50).

3. SENTENCES WITH MULTIPLE QUANTIFIED NPs: AN INTRODUCTION

In the previous sections, we saw what was new about generalized quantifiers, how one may extend quantification theory derived from systems based on first order logic. Now we consider the matter of interpreting sentences involving multiple generalized quantifiers corresponding to noun phrases (NPs).

\begin{align*}
(51) \quad \text{Exactly half the boys kissed some girl.}
\end{align*}

Examples of this type involve what is known as scope ambiguity. (51) is ambiguous, even though the ambiguity is neither a result of lexical ambiguity nor surface syntactic ambiguity (in both interpretations, kiss is the main transitive verb, exactly half the boys is the subject of the sentence and some girl is the object). To see the ambiguity, note first the following: the two interpretations corresponding to this sentence can be described as follows:

\begin{align*}
(52) \quad \text{a. Exactly half the boys kissed some girl or other (the other half didn’t kiss any girl).} \\
\quad \text{b. There is some girl, say Fatima, whom exactly half the boys kissed.}
\end{align*}
The first thing to note is that the two interpretations are logically independent, i.e., neither entails the other. In other words, it is possible for (52a) to be true in a situation and (52b) false; and for (52b) to be true and (52a) false. To see this, consider two situations M1 and M2. In M1, we have four boys Aaron, Brad, Chris and Dominique, and four girls Elisa, Fatima, Georgina and Harriet. Aaron kissed Elisa, Brad kissed Fatima, Chris and Dominique didn’t kiss anyone. In M1, (52a) is true, but (52b) is false, since there is no girl whom exactly half the boys kissed. In M2, we have the same boys and girls, but Aaron and Brad kissed Fatima, Chris kissed Harriet, Dominique didn’t kiss anyone. In M2, (52b) is true since there is one girl, namely, Fatima, who was kissed by exactly half the boys, but there are three boys who are girl-kissers: hence more than half the boys, making (52a) false. How then are we to account for the two interpretations of the sentence?

There are two broad families of approaches that attempt to account for scope ambiguity. One assumes that the semantic composition rules must be formulated in such a way as to make it possible to assign more than one interpretation to sentences containing more than one NP. Another approach assumes that one must assume that rules of semantic composition apply not to surface syntax, but rather to a level of more abstract syntax, called Logical Form (LF). We will follow a version of the latter approach as it allows us to keep a relatively simple semantic theory, though it complicates the syntax somewhat. The choice between these two families of approaches is a complicated issue and involves empirical considerations that are beyond the scope of this article.

4. LOGICAL FORM (LF)

We assume, following the proposals of the linguist Robert May (May 1977), that NPs can move at a level of abstract syntax, leaving coindexed traces, just as overt syntactic movement does in languages that have them (e.g., *wh*-movement in English and many other languages). The difference is that whereas overt movement is visible to the phonology (displaced constituents are pronounced in a different position), movement in the abstract syntax is not; making a difference only to the interpretation.
This yields a level of abstract syntax called “Logical Form” (LF), and the movement rule responsible is often called “quantifier raising” (QR), or covert movement. So for sentences like (51), the relative position of the two NPs at LF will correspond to two distinct interpretations. There are two possible LFs for (51):

\begin{align*}
\text{a. } & [S'_1 [\text{exactly half the boys}]_1 [S'_2 [S'_3 [S_t_1 \text{ kissed } t_2 ]]]] \\
\text{b. } & [S'_2 [\text{some girl}]_2 [S'_3 [\text{exactly half the boys}]_3 [S'_5 [S_t_1 \text{ kissed } t_2 ]]]]
\end{align*}

One may note that a sentence with n NPs can, in principle have up to n! LFs. As is customary in semantic theory, structures are interpreted relative to models, which consist of a universe of discourse and a valuation function which assigns values to the lexical items of the language, constrained by the grammar of the language under investigation. We also need assignment functions (these being functions that assign values chosen from the universe of discourse to variables). In the context of natural language semantics, these will be the traces and pronouns. (Here we don’t look at pronouns and just stick to traces.) Truth is defined relative to a model M and an assignment function g and then relative to M. Semantic relationships like entailment, logical equivalence, etc. are then defined by generalizing over models.

5. RULES OF SEMANTIC COMPOSITION

A Model $M = \langle U, V \rangle$ contains a universe of discourse $U$ and a valuation function $V$ that assigns values to lexical items, constrained by the grammar. Thus, transitive verbs are assigned sets of ordered pairs of members of $U$, intransitive verbs and common nouns are assigned sets of members of $U$, proper names are assigned members of $U$ and determiners are assigned functions from sets of members of $U$ to sets of sets of members of $U$, as explained in previous sections. This can be illustrated with the following examples:

\begin{align*}
\text{a. } & \llbracket \text{boys } ]^M = \{ x \in U : x \text{ is a boy in } M \} \\
\text{b. } & \llbracket \text{girl } ]^M = \{ x \in U : x \text{ is a girl in } M \} \\
\text{c. } & \llbracket \text{kissed } ]^M = \{ \langle x, y \rangle : x \in U \text{ and } y \in U \text{ and } x \text{ kissed } y \text{ in } M \}
\end{align*}
d. $[[ \text{sings} ]]^M = \{ x \in U : x \text{sings in } M \} $

e. $[[ \text{exactly half the } ]]^M(A) = \{ B \subseteq U : |A \cap B| = \frac{1}{2} |A| \}$

(here $A \subseteq U$)

f. $[[ \text{some } ]]^M(A) = \{ B \subseteq U : |A \cap B| \neq 0 \}$

g. $[[ \text{every } ]]^M(A) = \{ B \subseteq U : A \subseteq B \}$

h. $[[ \text{most } ]]^M(A) = \{ B \subseteq U : |A \cap B| > \frac{1}{2} |A| \}$

One could extend this to other semantic determiners that form the basis for generalized quantifiers, following the observations in the previous sections. The rules of semantic composition can now be stated simply as follows:

$$(55) \begin{align*}
\text{a. If } \beta \text{ is a lexical item, } [[ \text{ } \beta^M ]]^M,g &= V(\beta) \\
\text{b. If } \beta \text{ is a trace or a pronoun } [[ \text{ } \beta^M ]]^M,g &= g(\beta) \\
\text{c. } [[ \text{ } \text{VP}_t \text{NP}^M ]]^M,g &= \{ x \in U : <x, [[ \text{NP}^M ]]^M,g > \in [[ \text{VP}_t^M ]]^M,g \} \\
\text{d. } [[ \text{ } \text{s NP VP}^M ]]^M,g &= 1 \text{ iff } [[ \text{NP}^M ]]^M,g \in [[ \text{VP}^M ]]^M,g \\
\text{e. } [[ \text{ } \text{s NP}_i \text{S'}^M ]]^M,g &= 1 \text{ iff } \{ x \in U : [[ \text{S'}^M ]]^M,g[x/e_i] = 1 \} \in [[ \text{NP}^M ]]^M,g 
\end{align*}$$

In (55), we make use of the notation of a modified assignment function $g[x/e_i]$ defined as that assignment function that is exactly like $g$ except possibly in what it assigns to $e_i : g[x/e_i](e_i) = x$. In other words:

$$(56) \begin{align*}
g[x/e_i](e_i) &= x \\
\text{for all } y \neq e_i, g[x/e_i](y) &= g(y) 
\end{align*}$$

Various logical notions can now also be defined as follows:

$$(57) \begin{align*}
\text{a. A logical form (LF) } \Delta \text{ is said to be true in a model } M \\
\text{iff } \Delta \text{ is true in } M \text{ w.r.t. all assignment functions.} \\
\text{b. } \Delta \text{ entails } \Omega \text{ iff in every model in which } \Delta \text{ is true, } \Omega \text{ is true as well.} \\
\text{c. } \Delta \text{ and } \Omega \text{ are logically equivalent iff they are true in exactly the same models.} \\
\text{d. } \Delta \text{ is valid iff it is true in all models.} \\
\text{e. } \Delta \text{ is contradictory iff it is false in all models.} 
\end{align*}$$
(As an exercise, the reader may wish to consider the models $M_1$ and $M_2$ given in the introduction. Assume $M_1 = \langle U_1, V_1 \rangle$ and $M_2 = \langle U_2, V_2 \rangle$ such that $U_1 = U_2 = \{\text{Aaron, Brad, Chris, Dominique, Elisa, Fatima, Georgina, Harriet}\}$

$V_1(\text{boy}) = V_2(\text{boy}) = \{\text{Aaron, Brad, Chris, Dominique}\}$

$V_1(\text{girl}) = V_2(\text{girl}) = \{\text{Elisa, Fatima, Georgina, Harriet}\}$

$V_1(\text{kissed}) = \{<\text{Aaron, Elisa}>, <\text{Brad, Fatima}>\}$

$V_2(\text{kissed}) = \{<\text{Aaron, Fatima}>, <\text{Brad, Fatima}>, <\text{Chris, Harriet}>\}$

Compute the semantic values for the two LFs in (53), and show that they conform to the expected outcome outlined in the introduction. Start with any assignment function of one’s choice, and show that the result does not depend on the assignment function one starts with. End of excursus).

6. A SEMANTIC UNIVERSAL OF DETERMINER MEANINGS IN LANGUAGE: CONSERVATIVITY

Having outlined how one may interpret sentences in a natural language containing generalized quantifiers, one may now ask if there are general semantic properties of natural language determiners that are of interest from a semantic point of view. We start with one property that has been proposed as a universal by Barwise and Cooper (1981): the property of Conservativity.

(58) A determiner $D$ is said to be conservative (CONS) iff

$$D(\text{A})(\text{B}) = D(\text{A} \cap \text{B})$$

It is easy to see that all the natural language determiners (simple and complex) we have seen so far are CONS. Compare the a. and b. sentences below and one can check that they are logically equivalent, illustrating the fact that they are conservative.

(59) a. Every student came to the party.
    b. Every student is a student who came to the party.

(60) a. Some student came to the party.
    b. Some student is a student who came to the party.
(61) a. At least n students came to the party.
b. At least n students are students who came to the party.

(62) a. Most students came to the party.
b. Most students are students who came to the party.

(63) a. Many students came to the party.
b. Many students are students who came to the party.

(64) a. At most n students came to the party.
b. At most n students are students who came to the party.

(65) a. Few students came to the party.
b. Few students are students who came to the party.

One can multiply examples, but it might be worthwhile, to see the point being made here, to consider what it would mean for a determiner to not be conservative. Consider a hypothetical determiner D1 whose meaning is given by the following condition:

(66) D1(A)(B) iff B ⊆ A

To see that D1 does not satisfy the condition CONS outlined in (58), one may simply note that D1(A)(A ∩ B), by (66) iff A ∩ B ⊆ A. A ∩ B ⊆ A is a tautology, and so is always true, no matter what. But (66) is a contingent statement, sometimes true and sometimes not. Therefore, whenever (66) is false, D1(A)(A ∩ B) is still true. In fact, there is one natural language expression that looks very suspiciously like D1 in (66): this is the English expression only (and its counterparts in other languages). Suppose I say only students are allowed in the hall, I mean that the people allowed in the hall are students, but only students are students who are allowed in the hall cannot be false, ever. Does this falsify the conservativity universal? Not quite: it turns out that only is not a true determiner, though it suspiciously looks like one in sentences like only students are allowed in this hall. One can see this by examining the distribution of only. Only is a focus-sensitive adverbial that can attach to all kinds of constituents, including verbs, verb phrases, and also full NPs, and so is not a determiner.

(67) a. John only SANG. (focus on “sang”)
b. John only sang SONATAS. (focus on “sonatas”)
c. Only the VIRTUOUS shall enter the kingdom of heaven.
(67c) shows clearly that only is not a determiner. There are other expressions like only (e.g., mostly) that can also look like determiners at first, but turn out to be focus sensitive adverbials and so exempt from the conservativity universal in sentences where they appear to be so.

One may also construct hypothetical determiners that violate CONS, and as expected, never appear in natural language. One such example, due to the linguist Richard Larson, is a hypothetical determiner NALL, whose semantics is given in (68).

\[(68) \text{NALL}(A)(B) \text{ iff } U-A \subseteq B\]

If such a determiner NALL were to exist, Nall squares are striped would mean something like everything that is not a square is striped. It is easy to check that NALL is not CONS, in the following model:

\[(69) U = \{a, b, c, d\} \]
\[A = \{a, b\} \]
\[B = \{a, c, d\} \]

In (69), U-A = {c, d}, and so U-A \subseteq B. But A \cap B = \{a\}, and so U-A is not a subset of A \cap B. This means that NALL(A)(B) is true in (69), but NALL(A)(A \cap B) is not. Determiners like NALL do not exist in natural language.

There are various interesting consequences of the CONS universal. Work by Keenan, Stavi and others (see, e.g., Keenan and Stavi 1986) have shown, e.g., that the CONS universal considerably reduces the space of possible determiners. One can also show that Conservativity is preserved under Boolean combinations like conjunction, disjunction and complementation. This means that forming complex determiners by adding and, or and not to conservative determiners will only yield other conservative determiners, and so on.

7. INTERSECTIVITY, SYMMETRY AND EXISTENTIALITY

Let us now consider another property of determiners that is definitely not a universal. However, this property picks out a class of determiners whose membership in this class has grammatical consequences.
A determiner $D$ is said to be intersective (INT) iff $D(A)(B) = D(A \cap B)(B)$.

A determiner $D$ is said to be symmetric (SYM) iff $D(A)(B) = D(B)(A)$.

A determiner $D$ is said to be existential (EXIST) iff $D(A)(B) = D(A \cap B)(E)$

(In (72), $E$ is the universe of discourse, standing for “everything that exists”). It turns out that if $D$ is conservative, then (70)-(72) represent equivalent conditions. That is, assuming that $D$ is conservative, it will be intersective if and only if it is symmetric if and only if it is existential. We show this as a theorem (or, series of theorems).

**Theorem 1.** Let $D$ be CONS. Then:

(i) If $D$ is INT, it is SYM.

(ii) If $D$ is SYM, it is EXIST.

(iii) If $D$ is EXIST, it is INT.

From (i)-(iii) it follows that INT, SYM and EXIST are equivalent conditions under CONS.

Proof of (i): Let $D$ be INT. Then $D(A)(B) = D(A \cap B)(B) = D(A \cap B)(A \cap B)$ (by CONS and set theory). $D(B)(A) = D(B \cap A)(A) = D(B \cap A)(B \cap A)$ (by CONS and set theory) = $D(A \cap B)(A \cap B)$ (by set theory). Therefore, $D(A)(B) = D(B)(A)$. Therefore, $D$ is SYM.

Proof of (ii): Let $D$ be SYM. $D(A \cap B)(E) = D(A \cap B)(A \cap B)$ (by CONS and set theory)

$D(A)(B) = D(B)(A) = D(B)(A \cap B)$ (by CONS, set theory) = $D(A \cap B)(B)$ (by SYM) = $D(A \cap B)(A \cap B)$ (by CONS, and set theory). Therefore, $D(A)(B) = D(A \cap B)(E)$. Therefore, $D$ is EXIST.

Proof of (iii): Let $D$ be EXIST. Therefore, $D(A)(B) = D(A \cap B)(E) = D(A \cap B)(A \cap B)$ (by CONS and set theory). $D(A \cap B)(B) = D(A \cap B)(A \cap B)$ (by CONS and set theory). Therefore, $D(A)(B) = D(A \cap B)(B)$. Therefore, $D$ is INT.
Now, which determiners are INT/SYM/EXIST and which are not? It turns out that these three (equivalent) properties pick out a class of determiners which had earlier in the syntactic literature been called weak determiners. Determiners like some, a, many, few, a few, numerals, etc. appear freely in a construction called the existential construction:

(73) a. There are some rabbits in the garden.
    b. There is a rabbit in the garden.
    c. There are few/a few/many rabbits in the garden.
    d. There are at least/exactly/at most n rabbits in the garden.
    e. There are no rabbits in the garden.

Milsark (Milsark 1974, 1977) called the NPs that appear in the existential construction, weak NPs (NPs that result from combining weak determiners with nominal material). These can be contrasted with proper names, definite NPs and determiners like every, which are not allowed in the existential construction.

(74) a. * There is every rabbit in the garden.
    b. * There are both rabbits in the garden.
    c. * There is the rabbit in the garden.
    d. * There is John in the garden.

NPs that are not allowed in the existential construction are called strong NPs. Another interesting observation of Milsark’s was that the determiners many and few had two readings each: a strong reading and a weak reading, and that the existential construction disambiguates the two readings by allowing only the weak reading, and disallowing the strong reading. Thus, many people did well on the exam could mean that the number of people doing well on the exam was sufficiently large (the weak, cardinal reading) or that the number of people doing well the exam was a sufficiently large proportion of the number of people (the strong, proportional reading). Milsark observed that the existential construction only allowed the weak, or cardinal, reading.

Barwise and Cooper, and later Keenan, extending the observation of Barwise and Cooper (see Keenan 1987), noted that the class of NPs allowed in the existential construction are exactly those that contain as
specifier a determiner that is symmetric or existential. (This is hedging somewhat. Barwise and Cooper define notions called “positive strong”, “negative strong” and “weak” in a slightly different fashion, and then argue that the weak determiners are intersective. Keenan takes the intersectivity/existentiality properties as the starting point, and provides an account of existential statements on that basis.) This can be easily checked by noticing that *a*, *some*, *many* and *few* on one interpretation and the numerals are all symmetric determiners. Strong determiners like *every*, *the*, strong *many* and *few*, *both*, etc. are not. This is one of many examples of where a semantic property of a class of lexical items has grammatical consequences. (Another well-studied case is the case of Negative Polarity Items: see below). Barwise and Cooper relate the distribution of weak NPs in the existential construction to the hypothesis that the existential construction corresponds to the EXIST condition: on their terms, *there is/are D N’s in Y* means roughly that *D N’s in Y exist*, which in weak NPs is equivalent to *D N’s are in Y*. They also argue that with strong NPs, NPs in Y exist are either tautologies or contradictions, which make them strange and ultimately, ungrammatical. While tautologies and contradictions are not, in and of themselves, ungrammatical, recent work by Gajewski and others have argued that sentences of a particular form (which they specify theoretically) are ungrammatical when they express tautologies or contradictions, but others are not, and this is for principled reasons.

8. MONOTONICITY PROPERTIES OF DETERMINERS

Another property (or class of properties) that differentiates between semantic determiners that has grammatical consequences is monotonicity. Monotonicity properties of determiners are defined as follows:

A determiner D is *monotone decreasing* (↓) in the left argument if and only if for all A, B, C ⊆ U, if D(A)(C) and B⊆A, then D(B)(C).

A determiner D is *monotone increasing* (↑) in the left argument if and only if for all A, B, C ⊆ U, if D(A)(C) and A⊆B, then D(B)(C).
A determiner $D$ is *monotone decreasing* ($\downarrow$) in the right argument if and only if for all $A$, $B$, $C \subseteq U$, if $D(A)(C)$ and $B \subseteq C$, then $D(A)(B)$.

A determiner $D$ is *monotone increasing* ($\uparrow$) in the right argument if and only if for all $A$, $B$, $C \subseteq U$, if $D(A)(B)$ and $B \subseteq C$, then $D(A)(C)$.

Essentially, monotone increasing environments preserve subset to superset inferences, whereas monotone decreasing environments preserve superset to subset inferences. Examples of determiners which are monotone follow.

(75)  

a. *No*: Decreasing in both arguments ($\downarrow\downarrow$), as can be seen from

*No student snores* entails *No student of phrenology snores.*

*No student ate a vegetable* entails *No student ate a green vegetable.*

b. *Some*: Increasing in both arguments ($\uparrow\uparrow$), as can be seen from

*Some student of phrenology snores* entails *Some student snores.*

*Some student ate a green vegetable* entails *Some student ate a vegetable.*

c. *Every*: Decreasing in the first argument, increasing in the second argument ($\downarrow\uparrow$)

*Every student snores* entails *Every student of phrenology snores.*

*Every student ate a green vegetable* entails *Every student ate a vegetable.*

Monotonicity properties of determiners are grammatically relevant. It has been known for a long time that the distribution of a class of lexical items (words or idiomatic phrasal expressions), called *Negative Polarity Items* (NPIs, for short) is sensitive to the presence of a negative, or negative-like element. Examples of such expressions in English are *any* and *ever.*
It was shown by the semanticist William Ladusaw (Ladusaw 1979), building on earlier work by linguists like Fauconnier, that the best hypothesis for the distribution of NPIs is that such items are licensed in environments that preserve superset to subset inferences: these environments are called *Downward Entailing* (or DE) environments. The generalization that NPIs are licensed in DE environments is called the **Ladusaw-Fauconnier Generalization**. In the case of determiners, a DE environment is simply a downward monotone environment. The validity of the Ladusaw-Fauconnier Generalization can be seen from the fact that, for example, NPIs are licensed in both arguments of determiners like *no* or *few* (both (↓↓)), in neither argument of determiners like *some* or *many* (both (↑↑)), but for a determiner like *every* (↓↑), NPIs are licensed in the first argument but not in the second argument.

(77)  
| a. No student who has ever studied phrenology can fail this exam. |
| b. No student who has studied phrenology can ever fail the exam. |
| c. Few students who have ever studied phrenology can fail this exam. |
| d. Few students who have studied phrenology can ever fail the exam. |
| e. *Some student who has ever studied phrenology can fail this exam. |
| f. *Some student who has studied phrenology can ever fail the exam. |
| g. *Many students who have ever studied phrenology can fail this exam. |
| h. *Many students who have studied phrenology can ever fail the exam. |
| i. Every student who has ever studied phrenology will pass this exam. |
| j. *Every student who has studied phrenology will ever pass the exam. |
The topic of Polarity Sensitivity is an extremely complex one, and a full discussion of various issues related to the topic is beyond the scope of this article. Suffice it to say, however, that it is uncontroversial that monotonicity properties of determiners play an important role in this phenomenon in the grammar of natural languages.

9. LOGICALITY OF DETERMINERS

The question of what makes something a “quantifier” if one goes beyond the universal and existential quantifiers of first-order logic has occupied logicians and linguistic semanticists since the earliest days of the development of the theory of generalized quantifiers. The basic intuition is that “quantifiers” (in the sense of logic) do not care about the identity of the objects they quantify over, but rather, about the quantity of things they quantify over. In the case of unrestricted quantification, the universal quantifier cares about properties that all things share, the existential quantifier cares about properties that something has, the quantifier no cares about properties that no object has, and so on: the exact identity of the things that have the (relevant) properties is irrelevant, except indirectly. The intuition behind this can be formalized by means of the following definition.

Definition. Let $E$ and $E'$ be two universes such that $|E| = |E'|$. Let $m$ be an isomorphism from $E$ to $E'$, i.e., a function whose domain is $E$, is 1-1, and onto $E'$. For any set $A \subseteq E$, $m(A) = \{y \in E': \text{for some } x \in E, m(x) = y\}$. A determiner $D$ is then said to be “quantificational”, or logical (alternatively, called isomorphism-invariant, or ISOM, for short) if and only if for all $A, B \subseteq E$, $D_E(A)(B) = D_{E'}(m(A))(m(B))$.

Variants of the ISOM condition were first proposed in logic in the work of Mostowski (1957) for the case of unrestricted quantification, and Lindström (1966) for the case of restricted quantification. The first discussion of a variant of the ISOM condition in the linguistics literature can be found in the work of Higginbotham and May (Higginbotham and May 1981). The formulation of the condition in some of these works is in terms of a different, but related condition called “automorphism
invariance”, which is a weaker notion (all ISOM relations are automorphism invariant, but not vice versa), but this is not relevant in the discussion that follows. A couple of points to note here: first, in our previous discussion of generalized quantifiers, we had ignored the universe of discourse; however, it is clear that the truth of $D(A)(B)$ is relative to the universe of discourse $E$ that $A$ and $B$ are subsets of, and the relevance of the universe of discourse will become clear in the discussion that follows. Secondly, ISOM formalizes the idea that determiners that are “quantificational” or logical care not about the identity of the things that have a certain property but rather the quantities of things being quantified over. It essentially says that if you change the identity of the things being quantified over consistently (this is achieved by the 1-1 and onto character of the function), the truth of the quantified statement will not change.

The ISOM property has a couple of interesting consequences. If a determiner $D$ is ISOM, then one can show that $D_E(A)(B)$ is a function of four numbers: $|A-B|$, $|A \cap B|$, $|B-A|$ and $|E-(A \cup B)|$. Towards this end, we prove the following theorem.

**Theorem 2.** Let $D$ be ISOM. Let $E$ and $E'$ be two universes such that $|E| = |E'|$. Let $A,B \subseteq E$ and $A',B' \subseteq E'$ such that:

1. $|A-B| = |A'-B'|$
2. $|A \cap B| = |A' \cap B'|$
3. $|B-A| = |B'-A'|$
4. $|E-(A \cup B)| = |E'-(A' \cup B')|$

Then: $D_E(A)(B) = D_{E'}(A')(B')$

**Proof:** From (i)-(iv), it follows that there are four isomorphisms (1-1 and onto functions) of the following type:

- $m_1: A-B \rightarrow A'-B'$
- $m_2: A \cap B \rightarrow A' \cap B'$
- $m_3: B-A \rightarrow B'-A'$
- $m_4: E-(A \cup B) \rightarrow E'-(A' \cup B')$
Note that the domains of these four functions are pairwise disjoint and jointly exhaustive (i.e., the union of the domains is E’). Similarly, the ranges of these four functions are pairwise disjoint and jointly exhaustive (i.e., the union of the ranges is E’). This means that the relation \( m = m_1 \cup m_2 \cup m_3 \cup m_4 \) is a function, is 1-1 and onto with domain E and range E’. (This is because it would fail to be a function only if \( m \) mapped something, say, \( a \), in E to two different things in E’, but this could happen only if \( a \) were simultaneously in two different blocs in \( \text{dom}(m) \), but that is impossible because the blocs \( \text{dom}(m_1), \text{dom}(m_2), \text{dom}(m_3), \text{dom}(m_4) \) are pairwise disjoint. It would fail to be 1-1 only if two different things in \( \text{dom}(m) \) were mapped to the same thing, say, \( b \) in E’, but this could only happen if \( b \) were simultaneously in two different blocs in \( \text{ran}(m) \), but that is impossible because the blocs in the \( \text{ran}(m) \) are pairwise disjoint. The function \( m \) is obviously onto because the ranges of the four functions are jointly exhaustive.)

Now, note that \( A = (A-B) \cup (A \cap B) \), and therefore, \( m(A) = m(A-B) \cup m(A \cap B) = m_1(A-B) \cup m_2(A \cap B) = (A'-B') \cup (A' \cap B') = A' \). Similarly, \( B = (B-A) \cup (A \cap B) \), and by parallel reasoning, \( m(B) = B' \). Given that \( m: E \to E' \) is an isomorphism, and that \( D \) is ISOM, it follows that \( D_{E}(A)(B) = D_{m(E)}(m(A))(m(B)) = D_{E'}(A')(B') \). QED.

We just saw that the property of logicality makes determiners functions of four cardinalities, viz., \( |A-B|, |A \cap B|, |B-A| \) and \( |E-(A \cup B)| \). This essentially guarantees that “quantificational” determiner meanings don’t care about the identities of the things quantified over, but just care about four quantities (in the restricted quantifier case, the ones we are interested in when doing natural language semantics). It turns out the space of possible logical determiner meanings can be reduced further, if one considers the fact that natural language determiners are CONS, and also the fact that most, and quite possibly, all determiners also satisfy an additional condition called the extension condition (EXT), defined as follows.

**Definition.** A determiner \( D \) is said to satisfy the **extension condition** (is EXT) if and only if for all \( A, B \subseteq E \subseteq E' \), \( D_E(A)(B) = D_{E'}(A)(B) \).
The extension condition says that the truth (or falsehood) of $D(A)(B)$ doesn’t change if one expands the universe of discourse. Notice that because $A, B \subseteq E$, $A \cup B \subseteq E$. So if $D$ is EXT, $D_E(A)(B) = D_{A \cup B}(A)(B)$, and so the extension condition has as one consequence (among others) that the truth of $D(A)(B)$ is not dependent on anything outside $A \cup B$. The common determiners we have seen so far are all EXT, and it’s quite possible that all determiners in natural language are EXT.

An example of a hypothetical determiner that fails EXT is Richard Larson’s NALL (see 68, 69 above), defined as

\begin{equation}
(78) \quad \text{NALL}_E(A)(B) = 1 \text{ iff } E - A \subseteq B
\end{equation}

Consider two universes $E \subseteq E'$, and $A$ and $B$ as follows

\begin{equation}
(79) \quad E' = \{a, b, c, d, e\} \\
E = \{a, b, c, d\} \\
A = \{a, b\} \\
B = \{a, c, d\}
\end{equation}

Now, $E - A = \{c, d\} \subseteq B$, and hence, $\text{NALL}_E(A)(B) = 1$. But, $E' - A = \{c, d, e\}$ is not a subset of $B$, and so $\text{NALL}_{E'}(A)(B) = 0$. We have seen before that NALL also violates CONS. One should remember, however, that CONS and EXT are logically independent conditions, i.e., it is possible for a determiner to be EXT but not CONS; and it is possible for a determiner to be CONS but not EXT.

An example of a (hypothetical) determiner that is EXT but not CONS is the determiner ONLY (this would be the denotation of the English \textit{only}, if it were a determiner), mentioned earlier in the article:

\begin{equation}
(80) \quad \text{ONLY}_E(A)(B) = 1 \text{ iff } B \subseteq A
\end{equation}

We have seen earlier that ONLY violates CONS. But it satisfies EXT, because if $E \subseteq E'$, $\text{ONLY}_E(A)(B) = 1 \text{ iff } B \subseteq A$, as the truth of ONLY-statements are only dependent on what’s in $A$ and $B$. Similarly, it is easy to construct hypothetical determiners that are CONS but violate EXT. Suppose we define a determiner meaning $D1$ as follows:

\begin{equation}
(81) \quad \text{D1}_E(A)(B) = 1 \text{ iff } |E - (A \cap B)| \leq 2
\end{equation}
D1E(A)(B) basically says that “at most two things are not both As as well as Bs”. So “D1 squares are striped” would mean “at most two things are not striped squares”. It is clear that such determiners do not exist. D1, however, is CONS, as D1E(A)(A ∩ B) = 1 iff |E-(A ∩ (A ∩ B))| ≤ 2 iff |E-(A ∩ B)| ≤ 2 iff D1E(A)(B) = 1, given (81). To see that D1 violates EXT, consider A, B ⊆ E ⊆ E’ as follows:

(82)  A = {a, b}
      B = {b, c}
      E = {a, b, c}
      E’ = {a, b, c, d}

Here, A ∩ B = {b}, and therefore E-(A ∩ B) = {a, c} and therefore |E-(A ∩ B)| ≤ 2, and so D1E(A)(B) = 1. But E’-(A ∩ B) = {a, c, d} and so |E’-(A ∩ B)| > 2, and hence, D1E(A)(B) = 0. D1, therefore, violates EXT.

Now suppose a determiner D satisfies ISOM, CONS as well as EXT. It is easy to show that D(A)(B) is now a function of just two cardinalities: |A-B| and |A ∩ B|. We now prove the following theorem.

**Theorem 3.** Let D be ISOM, CONS and EXT. Let E and E’ be two universes such that |E| = |E’|. Let A, B ⊆ E and A’, B’ ⊆ E’ be sets such that

(i)  |A-B| = |A’-B’|
(ii) |A ∩ B| = |A’ ∩ B’|

Then: D E(A)(B) = D E’(A’)(B’)

**Proof:** From (i)-(ii) it follows that there are 1-1 and onto functions m₁ and m₂:

m₁: A-B → A’-B’
m₂: A ∩ B → A’ ∩ B’

dom(m₁) ∪ dom(m₂) = (A-B) ∪ (A ∩ B) = A, and ran(m₁) ∪ ran(m₂) = (A’-B’) ∪ (A’ ∩ B’) = A’. Furthermore, dom(m₁) ∩ dom(m₂) = (A-B) ∩ (A ∩ B) = ∅, and ran(m₁) ∩ ran(m₂) = (A’-B’) ∩ (A’ ∩ B’) = ∅. This means that m = m₁ ∪ m₂ is a function, is 1-1, and dom(m) = A and ran(m) = A’ (see the reasoning in the Proof of Theorem 2). In other words, m: A → A’ is an
isomorphism such that \( m(A) = A' \), \( m(A-B) = m_1(A-B) = A' - B' \), and \( m(A \cap B) = m_2 (A \cap B) = A' \cap B' \). Now, \( D_E(A)(B) = D_E(A)(A \cap B) \) (by CONS). Since \( A, A \cap B \subseteq A \subseteq E \), it follows that \( D_E(A)(A \cap B) = D_A(A)(A \cap B) \) (by EXT) = \( D_m(A)(m(A))(m(A \cap B)) \) (by ISOM) = \( D_A'(A')(A' \cap B') \) (by EXT, since \( A', A' \cap B' \subseteq A' \subseteq E' \)) = \( D_E(A')(B') \) (by CONS). This means that if one keeps \( |A-B| \) and \( |A \cap B| \) constant, \( D(A)(B) \) will stay the same for all \( A, B \) if \( D \) is CONS, ISOM and EXT. QED.

All of this means that logical determiners of linguistic interest are functions of \( |A-B| \) and \( |A \cap B| \) alone. This is obvious if we look at the truth conditions associated with the usual determiners we encounter:

\[(83)\]

a. every\((A)(B) \leftrightarrow A \subseteq B \leftrightarrow |A-B| = 0\)
b. some\((A)(B) \leftrightarrow |A \cap B| \neq 0\)
c. no\((A)(B) \leftrightarrow |A \cap B| = 0\)
d. at least \( n \) \((A)(B) \leftrightarrow |A \cap B| \geq n\)
e. at most \( n \) \((A)(B) \leftrightarrow |A \cap B| \leq n\)
f. exactly \( n \) \((A)(B) \leftrightarrow |A \cap B| = n\)
g. the\(_{sg}\)(A)(B) \leftrightarrow |A \cap B| = 1 \text{ and } |A-B| = 0 \text{ (assuming a Russellian analysis of singular definite descriptions)}
h. both\((A)(B) \leftrightarrow |A \cap B| = 2 \text{ and } |A-B| = 0\)
i. between \( m \) and \( n \) \((A)(B) \leftrightarrow m \leq |A \cap B| \leq n\)
j. many\(_1\)\((A)(B) \leftrightarrow |A \cap B| \geq k\), where \( k \) is some contextually determined number (cardinal \textit{many})
k. many\(_2\)\((A)(B) \leftrightarrow |A \cap B|/(|A \cap B|+|A-B|) \geq k\), where \( k \) is some contextually determined number (proportional, or presuppositional \textit{many})
l. few\(_1\)\((A)(B) \leftrightarrow |A \cap B| \leq k\), where \( k \) is some contextually determined number (cardinal \textit{few})
m. few\(_2\)\((A)(B) \leftrightarrow |A \cap B|/(|A \cap B|+|A-B|) \leq k\), where \( k \) is some contextually determined number (proportional, or presuppositional \textit{few})
n. a few \((A)(B) \leftrightarrow |A \cap B| \geq k\), where \( k \) is some contextually determined small number.
o. most\((A)(B) \leftrightarrow |A \cap B| > |A-B|\)
p. the \( n \) \((A)(B) \leftrightarrow |A \cap B| = n \text{ and } |A-B| = 0\)
q. neither\((A)(B) \leftrightarrow |A-B| = 2 \text{ and } |A \cap B| = 0\)
The list can be further expanded. The astute reader will have noticed that many of these determiners are functions of just $|A \cap B|$. The class of logical determiners that have this characteristic is the class we have already met before: this is the class of Symmetric/Intersective/Existential determiners. We can now prove the following theorem.

**Theorem 4.** Let $D$ be ISOM, CONS, EXT and SYMM. Let $A, B \subseteq E$ and $A', B' \subseteq E'$ be sets such that $|A \cap B| = |A' \cap B'|$. Then, $D_E(A)(B) = D_E'(A')(B')$.

**Proof:** Because $|A \cap B| = |A' \cap B'|$, it follows that there is an isomorphism $m: A \cap B \rightarrow A' \cap B'$. It follows that:

$$
D_E(A)(B) = D_E(A)(A \cap B) \quad \text{(by CONS)} = D_E(A)(A \cap B) \quad \text{(by SYMM)} = D_E(A \cap B)(A \cap B) \quad \text{(by CONS and set theory)} = D_{A \cap B}(A \cap B)(A \cap B) \quad \text{(by EXT)} = D_{m(A \cap B)}(m(A \cap B))(m(A \cap B)) \quad \text{(by ISOM)} = D_{A' \cap B'}((A' \cap B')(A' \cap B')) = D_E(A' \cap B')(A' \cap B') \quad \text{(by EXT)} = D_E(A')(A' \cap B') \quad \text{(by CONS and set theory)} = D_E(A')(A' \cap B') \quad \text{(by SYMM)} = D_E'(A')(B') \quad \text{(by CONS)}. \quad \text{QED.}
$$

The fact that logical determiners that are CONS as well as EXT are functions of just $|A-B|$ and $|A \cap B|$ can be exploited to describe the logic of determiners for finite universes by means of semantic trees as follows: we take each point in the tree to represent two coordinates: the first coordinate $a = |A-B|$ and the second coordinate $b = |A \cap B|$. The points are arranged according to $a+b = |A|$ in increasing order, and each point gets a $+$ or a $-$ value depending on whether $D(A)(B)$ is true or false. As an illustration, the tree for *every* reads as follows:

| $|A|$ | $(0,0)+$ | $(0,1)+$ | $(1,0)-$ | $(0,2)+$ | $(1,1)-$ | $(2,0)-$ | $(0,3)+$ | $(1,2)-$ | $(2,1)-$ | $(3,0)-$ |
|---|---|---|---|---|---|---|---|---|---|---|

...
Similarly, the tree for *some* reads as follows:

\[ |A| = 0 \quad (0,0)- \]
\[ |A| = 1 \quad (0,1)+ (1,0)- \]
\[ |A| = 2 \quad (0,2)+ (1,1)+ (2,0)- \]
\[ |A| = 3 \quad (0,3)+ (1,2)+ (2,1)+ (3,0)- \]

…

The tree for *no*:

\[ |A| = 0 \quad (0,0)+ \]
\[ |A| = 1 \quad (0,1)- (1,0)+ \]
\[ |A| = 2 \quad (0,2)- (1,1)- (2,0)+ \]
\[ |A| = 3 \quad (0,3)- (1,2)- (2,1)- (3,0)+ \]

…

Work by Van Benthem, Westerståhl and others have shown that a variety of properties of logical determiners (e.g., monotonicity properties) can be “read off” of such semantics trees. The interested reader may wish to consult Peters and Westerståhl (2006) for a survey, and references therein for original work on this topic.

Before wrapping up this section, I would like to briefly discuss determiners that have been analyzed as semantic determiners that are not logical. The prime example of such a determiner, as argued in works such as Keenan and Stavi (1986), are possessives. Possessives in English and many other languages (but not all) are syntactically determiners. They are clearly not logical, as, e.g., the possessive *John’s* in *John’s book is 500 pages long* cares about the identity of *John’s book*: it is that particular thing that is being asserted to be 500 pages long. (The reader can convince themselves that *John’s* is not ISOM: an isomorphism \( m \) could assign the extension of *book* to, say a singleton A whose member is not a book and the extension of *is 500 pages long* to another set B; but if the object in A has no relationship to John, *John’s book is 500 pages long* could be true but *John’s A is a B* would be false even if the member of A is a member of B). Keenan and Stavi note, however, that possessives analyzed as determiners are CONS, since *John’s book is 500 pages long*
is logically equivalent to *John’s book is a book which is 500 pages long*. They take possessives to be one kind of basic CONS function in natural language and argue that the class of natural language determiners is the class of determiners that are Boolean combinations of basic determiners, including possessives.

A different way of looking at possessives is to view them as definite descriptions: this is a view that goes back to Russell and a modern version of the view can be found in Higginbotham (1983). The idea is to analyze *John’s book is 500 pages long* as being a standby for “the book that bears the POSS-relation to John is 500 pages long”. Here “POSS(x, y)” is a highly context-dependent relation that could mean x is in possession of y, or x is the author of y, or a whole bunch of interest-related relations whose exact value is determined by the context. (Suppose there is a competition where people read novels and are judged accordingly: here, *John’s book* might mean the book that John chose to read, for example.) Here, I consider the case where the nominal that the possessive restricts is singular (the plural case involves complications that I wish to ignore). With this in mind, let us say that

\[(84) \text{John’s (A)(B) } \leftrightarrow \text{the sg}(A \cap \lambda x \text{POSS(John, x)})(B)\]

The semantics in (84) can be guaranteed by assuming that ‘*s*’ is a morpheme whose translation in GQ is by means of a function which applies to y to yield the determiner which when applied to A yields a generalized quantifier, that is,

\[(85) \text{tr(‘*s*’) = } \lambda y \lambda A \lambda B \text{the } \text{sg}(A \cap \lambda x \text{POSS(y, x)})(B)\]

This analysis has various advantages: on the one hand, the meaning of the possessive is decomposed into a logical determiner the _sg_ (for which we assume a Russelian analysis) and a POSS relation which when it combines with a proper name (or an individual variable: this would be necessary in possessives like *every student’s*) yields a component of meaning that is not logical or “quantificational”. The analysis in (84) also predicts the apparent Conservativity of possessives. This is because by (84), John’s (A)(B) \( \leftrightarrow \text{the } \text{sg}(A \cap \lambda x \text{POSS(John, x)})(B)\). Let C =
\( \lambda x \text{POSS}(\text{John}, x) \). Then John’s \((A)(B) \leftrightarrow \text{the}_s (A \cap C)(B) \), and similarly, John’s \((A)(A \cap B) \leftrightarrow \text{the}_s (A \cap C)(A \cap B) \). Since the \(s\) is CONS, John’s \((A)(B) \leftrightarrow \text{the}_s (A \cap C) \leftrightarrow \text{the}_s (A \cap C)(A \cap C \cap B) \leftrightarrow \text{the}_s ((A \cap C)(A \cap B)) \) (by set theory) \( \leftrightarrow \text{the}_s (A \cap C)(A \cap B) \) (by CONS) \( \leftrightarrow \) John’s \((A)(A \cap B) \) (by definition). The CONS property of the possessive thus follows.

10. FURTHER RESEARCH QUESTIONS AND SUMMARY

In the above survey, I provided a basic introduction to the role played by the theory of Generalized Quantifiers in logic and natural language. Quantification is a live topic of research in both logic and natural language semantics. In linguistic semantics, there is a variety of topics that are currently under investigation: both related to the semantics proper as well as questions of the syntax-semantics map. One topic in the syntax-semantics interface has to do with the relative scope of quantifiers in sentences that contain multiple quantifiers. It turns out that not all relative scopes are attested with all quantifiers: they are subject to constraints. Polarity sensitivity and its relationship to monotonicity properties of determiners is another active area of research. Negative quantifiers pose special problems: thus, sentences containing the determiner no sometimes yields readings that apparently require splitting it into a negation and an existential determiner (the most natural interpretation of the English sentence you need go no further is that you do not need to go any further, rather than the two scopal readings you are required to not go any further and there is no further distance that you are required to go, which are the only two readings predicted if no is analyzed as we have done in this article.). A third active area of research has to do with the internal structure of semantic determiners: we have seen repeatedly in this article that what we call “semantic determiners” are syntactically complex, and have connections to a variety of other constructions in natural language: most commonly comparatives and superlatives. How the meaning of such syntactically complex determiners comes about is not always straightforward, and remains a hot topic of investigation. The interested reader can refer to works like Szabolcsi (2010) or Peters and Westerståhl (2006) for further reference.
REFERENCES


Utpal Lahiri is an Associate Professor at the Department of Linguistics and Contemporary English, EFL University, Hyderabad. His research interests are primarily in semantic theory, with its interfaces with syntax and other areas of theoretical linguistics. He has published widely on the semantics of questions, polarity, scalar particles and on their relevance to semantic theory in general, and is currently involved in research on modals.

Utpal Lahiri
lahiri1963@gmail.com