Philosophy of Language

Lecture 2: Reference and quantification

Zoltán Gendler Szabó
CreteLing 2019
Tuesday, July 16

1. Austere truth-conditional semantics

1. The truth and nothing but the truth

Given Compositionality and Truth-conditionality, one of the things meanings of words, phrases, and clauses do is help determine truth-conditions of declarative sentences in which those expressions occur.

On the most austere conception of semantics, they have no other job.

On this conception, we should think of sub-sentential meanings purely instrumentally. There are many alternative systems of geographic coordinates we could use to fix a position of a ship at sea and there are many equally good ways to pick semantic values and rules to fix the conditions under which declarative sentences are true.

1. Austere truth-conditional semantics

2. The normal and the warped

Austere truth-conditional semantics does not rely on a pre-theoretical notion of reference.

1. Austere truth-conditional semantics

2. The normal and the warped

Austere truth-conditional semantics does not rely on a pre-theoretical notion of reference.

Let π be a proxy function that maps Frege to Russell, Russell to Tarski, and Tarski to Frege. On the austere conception, Normal semantics is no better than Warped semantics .

Normal semantics

$[\![\text{Frege}]\!]^w$ = Frege
$[\![\text{Russell}]\!]^w$ = Russell
$[\![\text{Tarski}]\!]^w$ = Tarski
$[\![\text{walks}]\!]^w$ = {x : x walks in w}

Warped semantics

$[\![\text{Frege}]\!]^w$ = Russell
$[\![\text{Russell}]\!]^w$ = Tarski
$[\![\text{Tarski}]\!]^w$ = Frege
$[\![\text{walks}]\!]^w$ = {x : π^{-1}(x) walks in w}

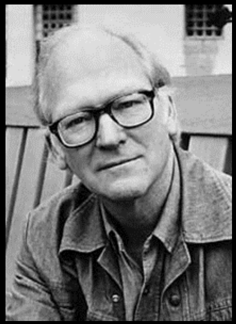
1. Austere truth-conditional semantics

3. Inscrutability

For Quine, the empirical basis of any theory consists of observation sentences. These are sentences of a language that linguistically competent and perceptually well-functioning speakers can come to agree on simply by witnessing a scenario. If these are all we base our theory choice on, the natural and warped semantics are evidentially on a par.

1. Austere truth-conditional semantics

3. Inscrutability



For Quine, the empirical basis of any theory consists of observation sentences. These are sentences of a language that linguistically competent and perceptually well-functioning speakers can come to agree on simply by witnessing a scenario. If these are all we base our theory choice on, the natural and warped semantics are evidentially on a par.

Davidson rejected the idea that there is a principled distinction between observation sentences and the rest, but he too believed that all evidence for or against a semantic theory comes from observable facts concerning the way speakers use their sentences—and this is enough for inscrutability.

1. Austere truth-conditional semantics

4. Appeal to simplicity

If we want to reject inscrutability we need a more liberal view about what counts as evidence.

A theory according to which, in normal cases, a particular use of a demonstrative pronoun refers to the object o the speaker demonstrates (usually by pointing, but often in some other way) is simpler than the one according to which it refers to some object o′ identified by first identifying o and then applying a proxy function.

Once inscrutability is given up for normal uses of demonstratives, we can leverage this to refute inscrutability for other expressions as well. We might insist, for example, that when someone introduces Frege by pointing at him and uttering Frege then the word uttered must refer to the individual demonstrated.

1. Austere truth-conditional semantics

5. Irrelevance?

Most semanticists believe that Russell is not the referent of Frege but insist that he, or any other artificial proxy of Frege, would serve equally well to model the real referent.

No theory should make assumptions beyond those it actually uses, and the assumption that semantic values are real world referents is idle in semantic theorizing (assuming all we want to capture are truth-conditions of sentences). Reference is not inscrutable, it is just beside the point.

But … if the semantic values of words and phrases are regarded as a more or less arbitrary tools to derive the correct truth-conditions for sentences then these values tell us very little about what words and phrases mean. That seems like a problem.

2. Referring expressions

1. The desiderata

Semantics textbooks usually tell us that referring expressions are pronouns, proper names, and definite descriptions.

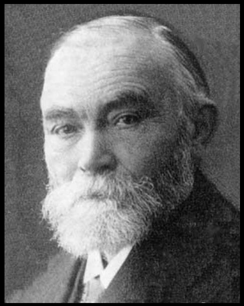
But bound or anaphoric pronouns, complex or descriptive proper names, and plural or mass definite descriptions are not always counted as referring expressions. There are also highly contentious examples of referring expressions; e.g. bare plurals, numerals, measure phrases.

Two desiderata:

i. Some expressions refer, some don’t. (E.g. Neptune refers but Vulcan does not.)
ii. Some expressions are for referring, some are not. (E.g. both Neptune and Vulcan are for referring but neither orbits the Sun nor Neptune orbits the Sun are for referring.)

2. Referring expressions

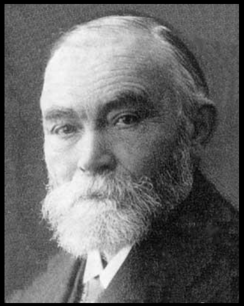
2. Fregean accounts



For Frege, every expression is a referring expression: the referent of orbits the Sun is a function from objects to truth-values and the referent of Neptune orbits the Sun is a truth-value. Vulcan has no referent but that is because it is not a real name, only appears to be one.

2. Referring expressions

2. Fregean accounts



For Frege, every expression is a referring expression: the referent of orbits the Sun is a function from objects to truth-values and the referent of Neptune orbits the Sun is a truth-value. Vulcan has no referent but that is because it is not a real name, only appears to be one.

e and t are basic types
if α and β are types, so is $\langle\alpha, \beta\rangle$

$\mathfrak{D}_e$ is a set of objects, $\mathfrak{D}_t$ the set of truth-values
$\mathfrak{D}_{\langle\alpha,\beta\rangle}$ is the set of functions from $\mathfrak{D}_\alpha$to $\mathfrak{D}_\beta$

If an expression of type α then its semantic value is a member of $\mathfrak{D}_\alpha$

2. Referring expressions

2. Fregean accounts

For Frege-inspired semanticists, referring expressions are those whose referent is an object (as opposed to a function). This helps with distinguishing referring expressions from other expressions but does not help with distinguishing referring expressions that refer from referring expressions that don't.

Saying that Vulcan refers to the null object does not help: if there is a null object it can be named (say, by ●) and we don't want ● = Vulcan to come out as true.

2. Referring expressions

3. Tarskian accounts

For Tarski, no expression is a referring expression: Neptune$_1$ is satisfied by a variable assignment g just in case g(x_1) is Neptune, Vulcan$_2$ is satisfied by a variable assignment g just in case g(x_2) is Vulcan, x_3 orbits the Sun is satisfied by a variable assignment g just in case g(x_3) orbits the sun.

2. Referring expressions

3. Tarskian accounts

For Tarski, no expression is a referring expression: Neptune$_1$ is satisfied by a variable assignment g just in case g(x_1) is Neptune, Vulcan$_2$ is satisfied by a variable assignment g just in case g(x_2) is Vulcan, x_3 orbits the Sun is satisfied by a variable assignment g just in case g(x_3) orbits the sun.

A variable assignment is a function that assigns an object to all variables. Satisfaction of a formula with respect to an assignment is defined recursively.

A sentence is true iff it is satisfied by some assignment. (Or, equivalently, a sentence is true iff it is satisfied by all assignments.)

2. Referring expressions

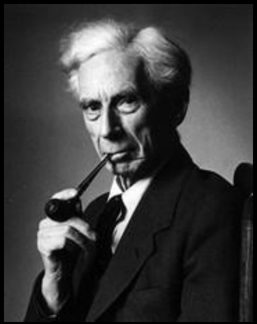
3. Tarskian accounts

For Tarski-inspired semanticists, referring expressions are all and only those that bear indices, and among referring expressions, the ones that actually refer are all and only the ones that are satisfied by some variable assignment.

The problem is that the category of referring expressions appears to be syntactically heterogeneous, and so, we have no obvious way to decide which expressions are supposed to bear indices.

2. Referring expressions

4. A Russellian suggestion



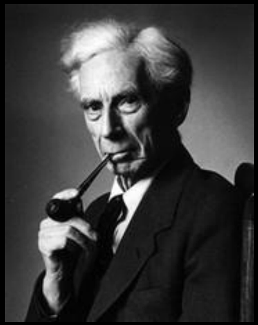
Russell thought logically proper names are mere tags – they designate their bearers without describing them. To understand a logically proper name we must be acquainted with its bearer.

Russellian acquaintance is demanding – we are only acquainted with ourselves and with our sense data. We can liberalize the notion to allow acquaintance with everyday objects, and we can extend it to allow acquaintance with things with which others are acquainted. Referring expressions are those one understands if and only if one is acquainted in this loose sense with a particular. (So, orbits the Sun and Neptune orbits the Sun are not referring expressions).

Vulcan is not a referring expression but it shares an important function with referring expressions—to designate a particular. It fails to perform this function—the definite description it abbreviates describes nothing at all.

3. Quantification

1. Russell on descriptions



On Russell's view, ordinary names are covert definite descriptions, and definite descriptions have the same sort of semantics as quantifiers.

C(everything) means that	C(x) is always true
C(something) means that	C(x) is sometimes true
C(all men) means that	if x is human then C(x) is always true
C(some men) means that	x is human and C(x) is sometimes true
C(a man) means what	C(some man) does
C(the man) means that	x is a human and for all y if y is a human then x=y and C(x) is sometimes true

3. Quantification

2. What is a quantifier?

In formal languages, quantifiers have the following characteristics:

Syntactically, quantifiers tend to be associated with a (single) variable. Semantically, the values of the variable are restricted (through its type) and its occurrences are bound within the scope of the quantifier (the formula immediately following the quantifier).

But none of these characteristics is universal: combinatorial logic expresses quantification without variables, first-order logic eschews type restrictions, and dynamic logic allows binding beyond the scope of the quantifier.

3. Quantification

3. Generalized quantifier theory: a proposal

Generalized quantifiers were first discussed by Andrzej Mostowski. According to generalized quantifier theory i. quantifiers belong to the semantic type $\langle\langle e,t\rangle,\langle\langle e,t\rangle,t\rangle\rangle$, and ii. are logical (i.e. their semantic values satisfy certain invariance conditions).

These requirements jointly guarantee that if Q is a quantifier then the truth-value of Q linguist(s) is/are asleep depends solely on a. how many linguists are asleep, b. how many linguists are not asleep, c. how many non-linguists are asleep.

In addition, iii. natural language quantifiers are conjectured to be conservative. This then guarantees that c. can be eliminated: the truth-value of Q linguist(s) is/are asleep depends solely on a. and b.

3. Quantification

3. Generalized quantifier theory: the problem

One problem with the view is that it implies that expressions like always and somehow are not quantifiers, which is rather implausible.

This might be fixed by allowing quantifiers in other semantic types as well. But presumably not in all, for then the definition would say that quantifiers are all and only the logical constants.

We would like an explanation why there are quantifiers in some semantic types but not others – e.g. why some is a quantifier while or is not.

Finally, the theory is not applicable to most formal languages.

3. Quantification

4. A (not so new) idea

Quantifiers are devices of generalization over their instances.

Instances are usually thought of as expressions obtained by swapping a quantifier for a proper name; generalizations over instances are sentences whose truth-value depends only on how many true and false instances it has.

3. Quantification

5. Substitutional clauses

Substitutional clauses for the universal and existential quantifiers within a language of first-order logic L can be given as follows:

$[\![\forall x.\, \varphi]\!]$ is true iff for every name $a \in L$, $[\![\varphi[x/a]]\!]$ is true
$[\![\exists x.\, \varphi]\!]$ is true iff for some name $a \in L$, $[\![\varphi[x/a]]\!]$ is true

3. Quantification

6. Truth-conditions in terms of substitutional instances

The instances of $Qx.\varphi$ relative to the unembedded occurrence of the quantifier Q are the formulae of the form $\varphi[x/a]$. An instance is positive if it is true; otherwise it is negative.

$[\![\forall x.\varphi]\!]$ is true iff the number of negative instances of $\forall x.\varphi$ relative to the unembedded occurrence of $\forall$ is zero.

$[\![\exists x.\varphi]\!]$ is true iff the number of positive instances of $\exists x.\varphi$ relative to the unembedded occurrence of $\exists$ is larger than zero.

3. Quantification

7. Objectual clauses

For substitutional clauses to deliver the intuitively correct truth-conditions we need to stipulate that everything within the model has exactly one name. Since this is unrealistic (especially if the model has infinitely many members) substitutional accounts of quantification have largely fallen out of favor.

What replaced them are objectual accounts:

$[\![\forall x.\varphi]\!]^g = 1$ iff for every object $o \in U$, $[\![\varphi]\!]^{g[x:o]} = 1$
$[\![\exists x.\varphi]\!]^g = 1$ iff for some object $o \in U$, $[\![\varphi]\!]^{g[x:o]} = 1$

These clauses cut out the middle man – the names paired with objects in a one-to-one fashion – and look directly at the objects in specifying truth-conditions.

3. Quantification

8. Truth-conditions in terms of objectual instances

The instances of $Qx.\varphi$ relative to the unembedded occurrence of the quantifier Q are ordered pairs of the form $\langle\varphi, o\rangle$, where o is an object that can be a value of x. $\langle\varphi, o\rangle$ is positive with respect to g just in case $[\![\varphi]\!]^{g[x:a]}$ is true; otherwise it is negative with respect to g.

The truth-conditions of quantified formulae can be specified in terms of objectual instances in the same way as in terms of substitutional instances:

> $[\![\forall x.\varphi]\!]^{g}$ is true iff the number of negative instances of $\forall x.\varphi$ relative to the unembedded occurrence of $\forall$ with respect to g is zero.
>
> $[\![\exists x.\varphi]\!]^{g}$ is true iff the number of positive instances of $\exists x.\varphi$ relative to the unembedded occurrence of $\exists$ with respect to g is larger than zero.

3. Quantification

9. Cardinality quantifiers

An expression ε is a cardinality quantifier iff whenever it occurs unembedded in a sentence σ, the truth-value of σ depends only on the cardinalities of the sets of its positive and negative instances relative to this occurrence of ε.

$\forall$ and $\exists$ are cardinality quantifiers, and so are quantifiers that can be defined in terms of them (e.g. $\exists_5$, $\exists_{<7}$etc.). The Rescher quantifier R (where $Rx.\varphi$ says that most things are φ) is not first-order definable, but still a cardinality quantifier.

3. Quantification

10. Instances in English

Let the expression ε occur unembedded in sentence σ and let δ be a demonstrative intersubstituible with ε. Then the instances of σ relative to ε are ordered pairs of the form $\langle\sigma[\varepsilon/\delta], o\rangle$ where o is some entity or other. An instance is positive with respect to the context c just in case $[\![\sigma[\varepsilon/\delta]]\!]^{c[\delta:o]}$ is true, and negative with respect to the context c just in case $[\![\sigma[\varepsilon/\delta]]\!]^{c[\delta:o]}$ is false. ($c[\delta: o]$ differs from c only in assigning o as a semantic value to δ.)

So, the instances of Every linguist is alseep relative to the occurrence of every are pairs of the form $\langle$That linguist is asleep, $o\rangle$. In positive instances o is a linguist who is asleep; in a negative instance o is a linguist who is not asleep. (I assume, as seems plausible, that in contexts where the demonstrative does not refer to a linguist That linguist is asleep is neither true nor false.) Every linguist is alseep is true iff the number of its negative instances is zero. So, every occurs here as a cardinality quantifier.

3. Quantification

11. Two predictions

Consider the sentence Jack used my coins to pay for coffee. Whether this sentence is true depends on which coins are mine, and so, its truth-value can change if the number of coins Jack used to pay for coffee and the number of coins he did not use to pay for coffee remains the same. The conclusion is that my does not occur in this sentence as a cardinality quantifier.

Whether Jack used only coins to pay for coffee is true depends on whether he also used something else, say, a dollar bill. The sentence can change its truth-value even if we hold the number of coins Jack used to pay for coffee and the number of coins he did not use to pay for coffee fixed. It follows that only does not occur in this sentence as a cardinality quantifier.

3. Quantification

12. Advantages

i. The account is broadly applicable to formal and natural languages – assuming we are willing to think of variables as devices of demonstration.
ii. It is cast in ordinary terms and – setting aside the precise definition of instances – it stays close to an intuitive idea.
iii. It explains why or is not a quantifier: disjunctions don't have instances, and so they cannot express generalizations over them.
iv. It leaves room for vague quantifiers: generalizations needn't be precise.
v. It can be generalized. Mass quantifiers (like much, or little) are not cardinality quantifiers but they can be characterized similarly by introducing ways of measuring (rather than counting) their instances.
vi. It captures the idea that quantifiers have something to do with quantity: they count (or measure) their true and false instances.

4. Summary

- The idea that reference is inscrutable is based on an overly narrow view of admissible linguistic evidence.
- We have no substantive and generally accepted account of what counts as a referring expression.
- The broadly accepted generalized quantifier view fails to provide an adequate account of what counts as a quantifier.
- Such an account is available: quantified sentences express generalizations over their instances. What counts as an instance varies from language to language.

the end (for now)