Names and Definite Descriptions

Larry Moss

Indiana University

North American Summer School on
Logic, Language, and Information
June 23-27, 2025

We add to $\mathcal{S}^\dagger$ names and (a very simple form of) definite descriptions.

This section adds names to $\mathcal{S}^\dagger$; definite descriptions come later.

We start with a set Names of names, and we use letters like a and b for names.
We add to the syntax of $\mathcal{S}^\dagger$ three kinds of sentences

$$\begin{array}{l} \mathsf{a} \text{ is } \mathsf{b} \\ \mathsf{a} \text{ isn't } \mathsf{b} \\ \mathsf{a} \text{ is a } \mathsf{p} \end{array} \tag{1}$$

In this last kind of sentence p may again be a literal (a noun or a complemented noun).
We don't need a sentence a isn't a p, since we have a is a $\overline{\mathsf{p}}$.
We call this language $\mathcal{S}^\dagger_{\mathsf{names}}$.

In a model $\mathcal{M}$, we interpret names by elements of the universe M. (These are sometimes called points.) For each name a, $[\![a]\!] \in M$. Here is the semantics:

$$
\begin{array}{lcl}
\mathcal{M} \models \mathsf{a\ is\ b} & \text{iff} & [\![\mathsf{a}]\!] = [\![\mathsf{b}]\!] \\
\mathcal{M} \models \mathsf{a\ isn't\ b} & \text{iff} & [\![\mathsf{a}]\!] \neq [\![\mathsf{b}]\!] \\
\mathcal{M} \models \mathsf{a\ is\ a\ p} & \text{iff} & [\![\mathsf{a}]\!] \in [\![\mathsf{p}]\!]
\end{array}
$$

The last point above is for all basic nouns p.
For its complement $\overline{\mathsf{p}}$, it follows that

$$\mathcal{M} \models \mathsf{a\ is\ a\ }\overline{\mathsf{p}} \quad \text{iff} \quad \mathcal{M} \not\models \mathsf{a\ is\ a\ p}$$

We extend the notion of a semantic negation:

φ	$\overline{\varphi}$
a is a p	a is a $\overline{\text{p}}$
a is b	a isn't b
a isn't b	a is b

The logic $\mathcal{S}^{\dagger}$

The letters p and q denote literals (nouns or complemented nouns).

$$\frac{}{\text{All } p \text{ are } p}\ \textsc{axiom} \qquad \frac{\text{Some } p \text{ are } q}{\text{Some } p \text{ are } p}\ \textsc{some}_1 \qquad \frac{\text{Some } p \text{ are } q}{\text{Some } q \text{ are } p}\ \textsc{some}_2$$

$$\frac{\text{All } p \text{ are } n \quad \text{All } n \text{ are } q}{\text{All } p \text{ are } q}\ \textsc{barbara} \qquad \frac{\text{All } q \text{ are } n \quad \text{Some } p \text{ are } q}{\text{Some } p \text{ are } n}\ \textsc{dar}$$

$$\frac{\begin{array}{c}[\varphi]\\ \vdots\\ \bot\end{array}}{\varphi}\ \textsc{raa}$$

$$\frac{}{\text{a is a}}\ \textsc{ref} \qquad \frac{\text{a is b}}{\text{b is a}}\ \textsc{sym} \qquad \frac{\text{a is b} \quad \text{b is c}}{\text{a is c}}\ \textsc{trans}$$

$$\frac{\text{a is a p} \quad \text{a is b}}{\text{b is a p}}\ \mathrm{R}_1 \qquad \frac{\text{a is a p} \quad \text{All p are q}}{\text{a is a q}}\ \mathrm{R}_2$$

The letters a and b are names, and as before p and q are literals. In addition, we allow (RAA), using as a contradiction sentences a is a p and a is a $\overline{\text{p}}$.

Recall that the logic $\mathcal{S}^\dagger$ allows us to trigger (RAA) from pairs Some p are q and All p are $\overline{\mathsf{q}}$, where p and q are any literals. Moving to $\mathcal{S}^\dagger_{\mathsf{names}}$, we also allow (RAA) to be triggered from any pair φ and $\overline{\varphi}$ as above.

Example

On the left below, we show that a is a p, a is a q $\vdash$ Some p are q.

$$\dfrac{\mathsf{a\ is\ a\ q} \qquad \dfrac{\mathsf{a\ is\ a\ p} \quad [\mathsf{All\ p\ are\ }\overline{\mathsf{q}}]}{\mathsf{a\ is\ a\ }\overline{\mathsf{q}}}\ \mathrm{R}_2}{\mathsf{Some\ p\ are\ q}}\ \mathrm{RAA} \qquad \dfrac{\dfrac{\mathsf{a\ is\ a\ p} \quad [\mathsf{a\ is\ a\ }\overline{\mathsf{q}}]}{\mathsf{Some\ p\ are\ }\overline{\mathsf{q}}} \qquad \mathsf{All\ p\ are\ q}}{\mathsf{a\ is\ a\ q}}\ \mathrm{RAA}$$

By the way, we could have taken the example on the left as a rule, call it ($\mathrm{R}_{2.5}$), and then derived (R_2) from ($\mathrm{R}_{2.5}$). This is shown on the right, where the step in the upper-left is an application of ($\mathrm{R}_{2.5}$).

We make things more interesting by making further additions coming from definite descriptions.

So if a is a name and p is a literal, then we add sentences a is the p, and a isn't the p.

We call the resulting language $\mathcal{S}^{\dagger}_{\text{names,the}}$. And here is the semantics:

$$\begin{array}{lll} \mathcal{M} \models \text{a is the p} & \text{iff} & [\![\text{p}]\!] = \{[\![\text{a}]\!]\} \\ \mathcal{M} \models \text{a isn't the p} & \text{iff} & [\![\text{p}]\!] \neq \{[\![\text{a}]\!]\}. \end{array}$$

Please note that $\mathcal{M} \models$ a isn't the p doesn't imply that a is a p, unlike what you would imagine from a real conversation.

This is called a conversational implicature:
nobody would say "a isn't the p" unless it was already known or presupposed that a is a p.

We take the negation the way we do so that we have a semantic negation.

Given the choice of being more linguistically realistic vs. making our life easier in the logic, we are taking the second choice for now.

However, all of this will be reconsidered shortly
(and in this class, this reconsideration is the main thing).

$$\frac{\text{a is the p}}{\text{a is a p}}\ \mathrm{R}_4$$

$$\frac{\text{a is the p} \quad \text{b is a p}}{\text{a is b}}\ \mathrm{R}_5$$

$$\frac{\text{a is the p} \quad \text{Some p are q}}{\text{a is a q}}\ \mathrm{R}_6$$

$$\frac{\text{a is the p} \quad \text{All q are p} \quad \text{b is a q}}{\text{b is the q}}\ \mathrm{R}_7$$

$$\frac{\text{a is the p} \quad \text{a is b}}{\text{b is the p}}\ \mathrm{R}_8$$

$$\frac{\text{a is the p} \quad \text{a is the q}}{\text{All p are q}}\ \mathrm{R}_9$$

$$\frac{\text{a is the p} \quad \text{b is the } \overline{\text{p}} \quad \text{a is a q} \quad \text{b is a q}}{\text{All } \overline{\text{q}} \text{ are q}}\ \mathrm{R}_{10}$$

$$\frac{\text{a is the p} \quad \text{b is the } \overline{\text{p}} \quad \text{a is a } \mathrm{q}_1 \quad \text{a is a } \mathrm{q}_2 \quad \text{b is a } \overline{\mathrm{q}}_1}{\text{All } \mathrm{q}_1 \text{ are } \mathrm{q}_2}\ \mathrm{R}_{11}$$

Add a rule of universal generalization to the system:

$$\frac{\Gamma \cup \{\mathsf{c} \text{ is a } \mathsf{p}\} \vdash \mathsf{c} \text{ is a } \mathsf{q} \qquad \mathsf{c} \text{ does not occur in } \Gamma}{\Gamma \vdash \text{All } \mathsf{p} \text{ are } \mathsf{q}} \ \textsc{ug}$$

Adopting this would make (R_{10}) and (R_{11}) provable.

For that matter, we could also prove (AXIOM), (BARBARA), and (R_9).

Thus, we could replace these five rules with (UG) and the resulting system would be complete.

This would not simplify the completeness proof, and one would need to prove as lemmas the statements of the deleted rules.

The change might well complicate the proof search. (But I did not investigate this matter.)
In the other direction, the completeness theorem that we have shown implies that every instance of (UG) is provable.

A Proposal for a Logic of Accommodation and Inference in the Spirit of Neutral Free Logic

Our formulation of $\mathcal{S}^{\dagger}_{\text{names,the}}$ has sentences a is b, a isn't b, a is a p, and a is the p, where p is a literal, and a and b are names.

But it lacks sentences of several forms that we would want to include:

$$
\begin{array}{l}
\text{the p is the q} \\
\text{the p isn't the q} \\
\text{the q is a p}
\end{array}
\qquad (2)
$$

As before p and q are literals, so the last form of sentence in (2) includes the sentences the q is a $\overline{\text{p}}$.

We might want to include these because they are English sentences, and because the presentation of the syntax would be easier and less ad-hoc if we did so.

One way would be to straightforwardly extend what we have done:

$\mathcal{M} \models$ the p is the q	iff	$[\![p]\!] = [\![q]\!]$, and both are singleton sets
$\mathcal{M} \models$ the p isn't the q	iff	$[\![p]\!] \neq [\![q]\!]$, and both are singleton sets
$\mathcal{M} \models$ the q is a p	iff	$[\![q]\!] \in [\![p]\!]$, and $[\![q]\!]$ is a singleton set

However, there are well-known issues with this.

If $[\![p]\!]$ is not a singleton in a given model, then both of the first two sentences come out false.

The second sentence is not the semantic negation of the first.

In a model where $[\![p]\!]$ is not a singleton, we might not want to say that the sentences are false.
We might want them to be “undefined” or “anomalous”.

This is what we do here.

So the semantics and proof theory will be partial in an appropriate sense.

Incidentally, we are not presenting this material in order to advance a proposal of what sentences like (2) mean.

Our view is that this is a complicated matter that one should decide on before turning to logic.

We are not doing this here. Instead what we are doing is to make a proposal without arguing for it,
and then to work out what the logic should be and how the completeness theorem should go.

Our proposal is a free logic in that the definite descriptions need not denote.
To make life a little simpler, we will take the names a to always denote, and of course this choice could be revised.

Our proposal is a neutral free logic in that we shall take sentences like those in (2) to be undefined in some models.

DEFINITION

We define a logical language $\mathcal{S}^{\dagger}_{\mathsf{acc}}$ as follows. As before, we begin with a set N of nouns and a set Names of names.

An individual term is either a name a or a definite description the p. We use letters like t and u for individual terms.
We take all of the sentences in the language of this lecture, together with the sentences in (2). We call these items the sentences of $\mathcal{S}^{\dagger}_{\mathsf{acc}}$.

A definedness assertion $\mathsf{t}\downarrow$ for t an individual term, and also $\varphi\downarrow$ for φ a sentence of this language.
When we speak of sentences, we do not include the definedness assertions $\mathsf{t}\downarrow$ and $\varphi\downarrow$.

Definition

Given a model $\mathcal{M}$, and a term t, we define a relation $\mathcal{M} \Vdash \mathsf{t}\downarrow$ below.
The same figure also defines the overall semantic relation $\mathcal{M} \Vdash \varphi\downarrow$.

Thus $\mathcal{M} \Vdash \varphi\downarrow$ implies by definition that $\mathcal{M} \models \varphi\downarrow$.

The point of the definition is to formalize the intuition that a definite description the p is only sensible in a model when $[\![\mathsf{p}]\!]$ is a singleton.
If this fails, then every sentence that has the p inside will be undefined.
(Again, this might or might not be what one wants.)

$\mathcal{M} \Vdash \mathsf{a} \downarrow$		always
$\mathcal{M} \Vdash (\mathsf{the\ p}) \downarrow$	iff	$[\![\mathsf{p}]\!]$ is a singleton set
$\mathcal{M} \Vdash (\mathsf{t\ is(n't)\ u}) \downarrow$	iff	$\mathcal{M} \Vdash \mathsf{t} \downarrow$ and $\mathcal{M} \Vdash \mathsf{u} \downarrow$
$\mathcal{M} \Vdash \mathsf{t\ is\ a\ p} \downarrow$	iff	$\mathcal{M} \Vdash \mathsf{t} \downarrow$
$\mathcal{M} \Vdash (\mathsf{All\ p\ are\ q}) \downarrow$		always
$\mathcal{M} \Vdash (\mathsf{some\ p\ are\ q}) \downarrow$		always
$\mathcal{M} \Vdash \varphi$	iff	$\mathcal{M} \Vdash \varphi \downarrow$ and $\mathcal{M} \models \varphi$