Deriving Syntactic Properties of Arguments and Adjuncts from Neo-Davidsonian Semantics

Tim Hunter[⋆]

Department of Linguistics
University of Maryland

Abstract. This paper aims to show that certain syntactic differences between arguments and adjuncts can be thought of as a transparent reflection of differences between their contributions to neo-Davidsonian logical forms. Specifically, the crucial underlying distinction will be that between modifying an event variable directly, and modifying an event variable indirectly via a thematic relation. I note a convergence between the semantic composition of neo-Davidsonian logical forms and existing descriptions of the syntactic properties of adjunction, and then propose a novel integration of syntactic mechanisms with explicit neo-Davidsonian semantics which sheds light on the nature of the distinction between arguments and adjuncts.

This paper aims to show that certain syntactic differences between arguments and adjuncts can be thought of as a transparent reflection of differences between their contributions to neo-Davidsonian logical forms. Specifically, the crucial underlying distinction will be that between modifying an event variable directly, as **violently** and **yesterday** do in (1), and modifying an event variable indirectly via a thematic relation, as **b** and **c** do in (1).

(1) a. Brutus stabbed Caesar violently yesterday

b. $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(e, \mathbf{b}) \wedge \mathsf{Stabbee}(e, \mathbf{c}) \wedge \mathbf{violent}(e) \wedge \mathbf{yesterday}(e)]$

I note a convergence between the semantic composition of neo-Davidsonian logical forms and the mechanisms added by Frey and Gärtner [1] to Stabler's Minimalist Grammar (MG) formalism [2] to allow adjunction phenomena. Frey and Gärtner provide an accurate formal encapsulation of *what* the distinctive syntactic properties of adjunction are. This paper contributes a novel integration of syntactic mechanisms with explicit neo-Davidsonian semantics which sheds light on *why* these properties cluster together.

1 Two Classes of Words

Consider the sentence in (2) and the variants of it in (3–6).

[⋆] Thanks to Norbert Hornstein, Greg Kobele, Paul Pietroski, Amy Weinberg and Alexander Williams for helpful discussions related to this paper.

(2) Brutus stabbed Caesar

(3) a. Brutus stabbed Caesar violently
b. Brutus stabbed Caesar yesterday
c. Brutus stabbed Caesar violently yesterday
d. Brutus stabbed Caesar yesterday violently

(4) a. * Brutus stabbed Caesar Cassius
b. * Brutus stabbed Caesar Antony
c. * Brutus stabbed Caesar Cassius Antony

(5) a. Caesar stabbed Brutus
b. Brutus stabbed Cassius
c. Antony stabbed Caesar

(6) a. * Brutus stabbed
b. * stabbed Caesar
c. * stabbed

First, we can infer from (3) that there is some class of words, including 'violently' and 'yesterday', which can be boundlessly added to the sentence in (2) without affecting grammaticality, though no word of this class need be present. Let us call this class of words Class 1. We also note that each sentence in (3) implies the one in (2), and infer that any sentence including a word of Class 1 implies the sentence just like it but with that word removed; see Fig. 1.

The data in (4) indicates that there is also some other class of words, including at least 'Cassius' and 'Antony', which can not be added to the sentence in (2) without affecting grammaticality. Let us call this class of words Class 2.

Next, we can infer on the basis of (5) that 'Brutus' and 'Caesar' belong to a single class of words, since they can be interchanged without affecting grammaticality; and that this class must be Class 2, that of 'Cassius' and 'Antony'. We also discover a difference between Class 1 and Class 2: while interchanging two Class 1 words does not produce any obvious difference in meaning (compare (3c) and (3d)), interchanging two Class 2 words does (compare (2) and (5a)). Finally, (6) shows that Class 2 words also can not be *removed* without affecting grammaticality, just as (4) shows that they can not be added.

In sum we have discovered two classes of words with the following properties.

(7) Distributional properties:
a. Of Class 1 words, any number zero or greater can be present in a sentence constructed around 'stabbed'.
b. Of Class 2 words, exactly two must be present in a sentence constructed around 'stabbed'.

(8) Semantic properties:
a. When two Class 1 words are interchanged, no obvious difference in meaning results.

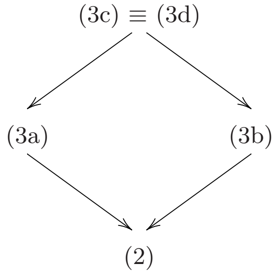

Fig. 1. The "diamond entailment" pattern among the sentences in (2) and (3).

b. When two Class 2 words are interchanged, an obvious difference in meaning results.[1]

c. Removing a Class 1 word from a sentence weakens its truth condition (i.e. never produces a false sentence from a true one, holding facts about the world constant).

These facts are not news, but the aim of this paper will be to show that the properties we have discovered for Class 1 words go together naturally, and that the properties we have discovered for Class 2 words go together naturally.[2] This provides an explanation for the lack of an imaginary Class 3, of which any number zero or greater can be present but which produce differences in meaning when interchanged, and the lack of an imaginary Class 4, of which exactly n must be present but which produce no difference in meaning when interchanged.

In section 2 I introduce an initially-appealing account of the distributional facts that leaves the semantic facts unexplained. After considering a way to capture the semantic facts alone in section 3, and a more complicated account of the distributional facts alone in section 4, I show in sections 5 and 6 that these two independent proposals can be integrated into a unified account of the underlying distinction between Class 1 and Class 2.

2 A First Approach

An account of the distributional properties in (7) of these two classes of words can be constructed in any formalism where expressions "select" others of certain types. Specifically, one can propose that 'stabbed' selects two words of Class 2

---

[1] The relevant sense of "interchange" here is to swap the occurrences of two lexical items throughout a *derivation*, not just to swap their linear order. Thus the synonymy of 'John gave a book to Mary' and 'John gave Mary a book' is not a counterexample.

[2] Of course, what I have called Class 1 is the class of adjuncts and what I have called Class 2 is the class of arguments. I avoid the terms "argument" and "adjunct" only to prevent familiarity with these properties from obscuring the fact that the observed correlations do not yet follow from anything.

to yield a complete sentential expression, and that a word of Class 1 selects an expression of some type to yield a new expression of that same type. In categorial grammars, this would be encoded using categories like those in (9) (ignoring directionality); in Minimalist Grammars (MGs) [2], it can be equivalently encoded using the lexicon in (10).

(9) stabbed :: $(v/d)/d$ Brutus :: d yesterday :: v/v
Caesar :: d violently :: v/v

(10) stabbed :: `=d =d v` Brutus :: `d` yesterday :: `=v v`
Caesar :: `d` violently :: `=v v`

This clearly derives the desired distributional properties in (7). But this "saturation" conception of syntactic composition is standardly taken to be reflected transparently in semantic composition, such that the syntactic selector denotes a function which is applied to the denotation of the selected expression — explicitly throughout the categorial grammar literature, including categorial treatments of MGs [3–5], and (with some exceptions in cases more complex than anything considered here) by Kobele [6, 7] for MGs as standardly presented. This produces the logical forms in (2′) and (3′) for the sentences in (2) and (3), which leave the semantic properties noted in (8) completely mysterious.[3] Why should (3c) and (3d) be synonymous, and why should each of the sentences in (3) imply the one in (2)?

(2′) **stab**(**c**)(**b**)

(3′) a. **violently**(**stab**(**c**)(**b**))
b. **yesterday**(**stab**(**c**)(**b**))
c. **yesterday**(**violently**(**stab**(**c**)(**b**)))
d. **violently**(**yesterday**(**stab**(**c**)(**b**)))

Of course, it is possible to define **yesterday** and **violently** such that these implications do hold, but this would be a separate, stipulated property of each such lexical item. Nothing in the formalism precludes lexical items with syntactic and semantic types identical to those of 'yesterday' and 'violently' (and therefore identical distribution), which would produce a difference in meaning when interchanged — that is, nothing in the formalism precludes the imaginary Class 3 mentioned above.

A second problem with this approach is that the selecting expression is generally taken to contribute the head of a newly-formed expression (such that

---

[3] To simplify the comparison with the addition of a specialised ADJOIN operation in MGs, discussed in section 4, I assume that 'yesterday' and 'violently' attach higher than both 'Brutus' and 'Caesar', such that semantically they must be operators on structured propositions. If they were to attach lower with a category of the form $(v/d)/(v/d)$, semantically VP-operators, the lack of the relevant implications remains.

'stabbed' is the head of 'stabbed Caesar'), which predicts that the head of 'Brutus stabbed Caesar yesterday' should be 'yesterday'. This conflicts with the general assumption that the head of 'Brutus stabbed Caesar yesterday' should be the head of 'Brutus stabbed Caesar'.[4] I return to this in section 4, after considering a way to capture the desired semantic properties in section 3.

3 Neo-Davidsonian Logical Forms

In order to capture the implications which are left unexplained by (2′) and (3′), it has been proposed [8–13] that the sentences in (2) and (3) should be associated with the logical forms in (2″) and (3″), where **stabbing**, **violent** and **yesterday** are predicates of events.[5]

(2″) $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(\mathbf{b})(e) \wedge \mathsf{Stabbee}(\mathbf{c})(e)]$

(3″) a. $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(\mathbf{b})(e) \wedge \mathsf{Stabbee}(\mathbf{c})(e) \wedge \mathbf{violent}(e)]$

b. $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(\mathbf{b})(e) \wedge \mathsf{Stabbee}(\mathbf{c})(e) \wedge \mathbf{yesterday}(e)]$

c. $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(\mathbf{b})(e) \wedge \mathsf{Stabbee}(\mathbf{c})(e) \wedge \mathbf{violent}(e) \wedge \mathbf{yesterday}(e)]$

d. $\exists e[\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(\mathbf{b})(e) \wedge \mathsf{Stabbee}(\mathbf{c})(e) \wedge \mathbf{yesterday}(e) \wedge \mathbf{violent}(e)]$

The logical form in (3d″) (equivalent to that in (3c″)) says that there exists an event with five properties: it was a stabbing event, its stabber was Brutus, its stabbee was Caesar, it was yesterday, and it was violent. The five words in (3d) correspond one-to-one with these properties. The observation in (8c) of the implications from (3) to (2) then follows trivially.

What is particularly crucial for present purposes is to note the differences between the way in which 'Brutus' and 'Caesar', and the way in which 'violently' and 'yesterday', contribute their respective conjuncts. Whereas the lexical contents of 'violent' and 'yesterday' modify the event variable directly, the lexical contents of 'Brutus' and 'Caesar' do so only indirectly via the thematic relations of Stabber and Stabbee. Thus there is no room for 'yesterday' to contribute to the meaning of (3d) in some way that is different from the way it contributes to the meaning of (3c), and likewise for 'violently'. But the more complex relation between 'Brutus' (or 'Caesar') and the event variable leaves room for 'Brutus' to contribute to the meaning of (5a) in a way that is different from the way it contributes to the meaning of (2).[6]

---

[4] Or, if 'yesterday' has a type like $(v/d)/(v/d)$, this approach predicts that the head of 'stabbed Caesar yesterday' should be 'yesterday'. The problem remains.

[5] It is more common to see something like $\mathsf{Stabber}(e, \mathbf{b})$ in place of $\mathsf{Stabber}(\mathbf{b})(e)$ as I have written here. For reasons that will become clear below, it is useful to think of Stabber and Stabbee as curried two-place functions — functions from individuals to event predicates (cf. [14]).

[6] Granted, the non-interchangeability observed in (8b) only follows if the thematic relations Stabber and Stabbee are necessarily distinct. The integrated proposal in

These logical forms therefore seem to permit the right two kinds of semantic contributions, given what we observed in section 1: elements that contribute atomic event predicates will be interchangeable in a way that leaves meaning unaffected (property (8a)), and those that contribute parts of complex event predicates will not (property (8b)). But what remains unexplained is why semantic property (8a) is correlated with distributional property (7a), and why semantic property (8b) is correlated with distributional property (7b).

As mentioned in section 2, it is possible to choose lexical meanings such that the function-application approach to semantic composition illustrated in ($2'$) and ($3'$) yields exactly the forms in ($2''$) and ($3''$), and therefore produces the desired implication relations. The meanings in (11) and (12), for example (with existential closure understood to apply to a sentential event predicate), would suffice.

(11) $$\mathbf{stab} = \lambda x \lambda y \lambda e.\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(y)(e) \wedge \mathsf{Stabbee}(x)(e)$$

(12) $$\mathbf{violently} = \lambda P \lambda e.P(e) \wedge \mathbf{violent}(e)$$

This correctly encodes the fact that ‘violently’ has the semantic properties we observed in section 1. It says nothing, however, about the general correlation between these semantic properties and the distributional properties of optionality and iterability (property (7a)). A formalism where semantics mirrors syntax in the way sketched in section 2 is equally consistent with other lexical items with the same type, and therefore the same distributional properties, as ‘violently’, but with different semantic properties. But we do not find variants with meanings like those shown in (13).[7]

(13) $$\begin{gathered}\mathbf{violently}' = \lambda P \lambda e.P(e) \vee \mathbf{violent}(e) \\ \mathbf{violently}'' = \lambda P \lambda e.P(e) \rightarrow \mathbf{violent}(e)\end{gathered}$$

In other words, a word with the distribution of ‘violently’ does not “choose” its own logical connective.

The same argument can be made with respect to the logical connective in the lexical meaning of **stab** given in (11): we do not find variants with meanings

---

section 5 will, I hope, make this seem reasonable. Even if it must be stipulated that they are distinct, this is certainly no worse than stipulating that $\mathbf{stab}(\mathbf{c})(\mathbf{b})$ and $\mathbf{stab}(\mathbf{b})(\mathbf{c})$ must be distinct in the system of section 2.

[7] This is of course a simplification: there exist adverbs with non-intersective interpretations, such as ‘allegedly’ and ‘apparently’. Whatever the correct logical form turns out to be for sentences such as ‘Brutus walked allegedly’, it seems likely that the relevant logical connective will be conjunction (and not, say, disjunction), although the conjuncts will not be of the simple sort I restrict attention to in this paper. Note that this problem is not confined to event-based adverbial modification: any theory that treats adjectival modification as generally intersective (eg. [15]) will encounter similar problems with ‘fake diamond’, ‘big ant’ and ‘alleged stabbing’. Larson’s [16] approach to some such adjectives treats them as intersections not of sets of individuals but rather of sets of events (see also [17]) — as mentioned above, even these seem to express conjunction of *something*.

like those shown in (14).

(14)
$$\begin{aligned}\mathbf{stab}' &= \lambda x \lambda y \lambda e.\mathbf{stabbing}(e) \vee \mathsf{Stabber}(y)(e) \vee \mathsf{Stabbee}(x)(e)\\ \mathbf{stab}'' &= \lambda x \lambda y \lambda e.\mathbf{stabbing}(e) \wedge \mathsf{Stabber}(y)(e) \rightarrow \mathsf{Stabbee}(x)(e)\end{aligned}$$

This relies crucially on the assumption that the logical form of 'Brutus stabbed Caesar' is not merely $\exists e[\mathbf{stabbing}(e, \mathbf{b}, \mathbf{c})]$; but see especially Schein [12, 18] for convincing evidence of this. Given this assumption, we can observe that verbs, like adverbs, do not "choose" their own logical connective.

In section 5 I will propose an alternative formalism which does not allow us to define the lexical meanings in (13) and (14), but rather permits only the degrees of freedom that natural languages seem to make use of. The new proposal will tie the neo-Davidsonian logical forms discussed here to an account of the syntactic properties of adjunction which I describe in the next section.

4 An Adjunction Operation in MGs

Recall that the simple selection-based proposal in section 2 wrongly establishes Class 1 words as heads — 'yesterday' as the head of 'Brutus stabbed Caesar yesterday', for example. I now turn to a proposed addition to the MG formalism which captures the distributional facts in (7) while avoiding this problem. This will be shown in the next section to unify elegantly with the event-based logical forms that capture the semantic properties in (8).

The operation for combining two expressions in the original MGs [2], MERGE, is the analogue of slash-elimination in categorial grammars that would derive (2) from the lexicon in (10) in the obvious way.[8]

(15)
$$\frac{\text{stabbed} :: \texttt{=d =d v} \qquad \text{Caesar} :: \texttt{d}}{\text{stabbed Caesar} : \texttt{=d v}}\ \textsc{merge}$$

This is a "symmetric feature checking" operation in that it deletes one feature from 'stabbed' and one feature from 'Caesar' in the above example.

Frey and Gärtner [1] introduce an *asymmetric* feature checking counterpart of MERGE, called ADJOIN. This checks a feature of a new sort, written `≈f`, on one of its input expressions, and leaves the features of the other input expression unchanged (though this other expression must have a feature `f` which "matches" the `≈f` feature to be checked).

(16)
$$\frac{\text{Brutus stabbed Caesar} : \texttt{v} \qquad \text{yesterday} :: \texttt{≈v}}{\text{Brutus stabbed Caesar yesterday} : \texttt{v}}\ \textsc{adjoin}$$

Furthermore, the ADJOIN operation is defined such that the head of the new expression is (the head of) the expression whose features remained unchanged. The

---

[8] Although it will not be crucial for an understanding of what is to come, a short introduction to the MG formalism is given in the appendix.

lexicon shown in (17) will therefore capture the desired distributional properties in (7) while assigning each expression the correct head.

(17) $\text{stabbed} :: \texttt{=d =d v} \qquad \text{Brutus} :: \texttt{d} \qquad \text{yesterday} :: \texttt{≈v}$
$\qquad\qquad\qquad\qquad\qquad\qquad \text{Caesar} :: \texttt{d} \qquad \text{violently} :: \texttt{≈v}$

As stated at the outset, these mechanisms provide an accurate formal description of the observed distributional properties, but the question remains: why does distributional property (7a) coincide with semantic property (8a), and distributional property (7b) with semantic property (8b)? Put differently, we have observed in section 3 that neo-Davidsonian logical forms permit the right two kinds of semantic composition (atomic event predicates and complex thematic ones), and also that the supplemented MG formalism permits the right two kinds of syntactic composition; we would still like to know why ADJOIN produces only simple event predicates and MERGE produces only complex ones. The next section answers this question by showing that the feature-checking patterns of ADJOIN and MERGE line up neatly with the differing patterns of semantic composition exhibited by Class 1 and Class 2 words respectively in neo-Davidsonian logical forms.

5 An Integrated Proposal

I note now a convergence between the neo-Davidsonian logical forms presented in section 3 and the MG formalism when supplemented with the ADJOIN operation presented in section 4. When ‘stabbed’ combines with ‘Caesar’ via the MERGE operation, as shown in (15), two features are checked, and the predicate of events that is introduced has two “ingredients”, Stabbee and **b**. When ‘Brutus stabbed Caesar’ combines with ‘yesterday’ via the ADJOIN operation, as shown in (16), only one feature is checked, and the predicate of events that is introduced has only one “ingredient”, **yesterday**. Suppose, then, that features are annotated with these semantic ingredients, as shown in (18).

(18) $\text{stabbed} :: \langle \texttt{=d}, \mathsf{Stabbee}\rangle \langle \texttt{=d}, \mathsf{Stabber}\rangle \langle \texttt{v}, \textbf{stabbing}\rangle$
$\text{Brutus} :: \langle \texttt{d}, \mathbf{b}\rangle \qquad \text{yesterday} :: \left\langle \texttt{≈v}, \textbf{yesterday}\right\rangle$
$\text{Caesar} :: \langle \texttt{d}, \mathbf{c}\rangle \qquad \text{violently} :: \left\langle \texttt{≈v}, \textbf{violent}\right\rangle$

The intuition is that **c** will be established as the Stabbee precisely because the application of MERGE that checks the `d` feature of ‘Caesar’ also checks the `=d` feature of ‘stabbed’ which is annotated with the Stabbee thematic relation. Likewise, **b** will be established as the Stabber as a result of the second application of MERGE. These two steps, constituting the derivation of (2), are shown in Fig. 2(a),[9] where we define $P \,\&\, Q := \lambda e.P(e) \wedge Q(e)$.

---

[9] When a sentential expression denotes a predicate P, the sentence’s truth condition is $\exists e[P(e)]$.

$$\dfrac{\text{Brutus} :: \langle \texttt{d}, \mathbf{b}\rangle \qquad \dfrac{\text{stabbed} :: \langle \texttt{=d}, \mathsf{Stabbee}\rangle\,\langle \texttt{=d}, \mathsf{Stabber}\rangle\,\langle \texttt{v}, \textbf{stabbing}\rangle \qquad \text{Caesar} :: \langle \texttt{d}, \mathbf{c}\rangle}{\text{stabbed Caesar} : \langle \texttt{=d}, \mathsf{Stabber}\rangle\,\langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}(\mathbf{c})\rangle}\,\textsc{merge}}{\text{Brutus stabbed Caesar} : \langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}(\mathbf{c}) \,\&\, \mathsf{Stabber}(\mathbf{b})\rangle}\,\textsc{merge}$$

(a) Two applications of MERGE with corresponding thematic roles being assigned

$$\dfrac{\text{Brutus stabbed Caesar} : \langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}(\mathbf{c}) \,\&\, \mathsf{Stabber}(\mathbf{b})\rangle \qquad \text{yesterday} :: \big\langle \texttt{≈v}, \textbf{yesterday}\big\rangle}{\text{Brutus stabbed Caesar yesterday} : \langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}(\mathbf{c}) \,\&\, \mathsf{Stabber}(\mathbf{b}) \,\&\, \textbf{yesterday}\rangle}\,\textsc{adjoin}$$

(b) An application of ADJOIN with corresponding conjunctive interpretation

$$\dfrac{\text{Brutus who is tall} : \langle \texttt{d}, \mathbf{b} \,\&\, \textbf{tall}\rangle \qquad \text{stabbed Caesar} : \langle \texttt{=d}, \mathsf{Stabber}\rangle\,\big\langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}^{\exists}(\mathbf{c})\big\rangle}{\text{Brutus who is tall stabbed Caesar} : \big\langle \texttt{v}, \textbf{stabbing} \,\&\, \mathsf{Stabbee}^{\exists}(\mathbf{c}) \,\&\, \mathsf{Stabber}^{\exists}(\mathbf{b} \,\&\, \textbf{tall})\big\rangle}\,\textsc{merge}$$

(c) An application of MERGE with existential closure of the selected predicate

Fig. 2. Derivations referred to in the main text

This suggests the general schema in (19) for the MERGE operation, though this will be revised in section 6.

(19)
$$\frac{s_1 :: \langle \mathtt{=f}_1, \theta_1\rangle \ldots \langle \mathtt{=f}_n, \theta_n\rangle\ \langle \mathtt{g}, \alpha\rangle \qquad s_2 :: \langle \mathtt{f}_1, \beta\rangle}{s_1 s_2 : \langle \mathtt{=f}_2, \theta_2\rangle \ldots \langle \mathtt{=f}_n, \theta_n\rangle\ \langle \mathtt{g}, \alpha \,\&\, \theta_1(\beta)\rangle}\ \textsc{merge}$$

The derivation can continue with an application of the ADJOIN operation which adds ‘yesterday’. This does not check any features of ‘Brutus stabbed Caesar’, and the conjunct added is simply **yesterday**, as shown in Fig. 2(b).

This suggests the general schema in (20) for the ADJOIN operation.

(20)
$$\frac{s_1 :: \langle \mathtt{f}, \alpha\rangle \qquad s_2 :: \left\langle \mathtt{\approx f}, \beta\right\rangle}{s_1 s_2 : \langle \mathtt{f}, \alpha \,\&\, \beta\rangle}\ \textsc{adjoin}$$

Clearly this requires that α and β be predicates of some common type. Let $\mathcal{T}$ be a function mapping syntactic categories to semantic types; and let e be the semantic type of entities/individuals, s the semantic type of events, and t the semantic type of truth values. Then we can set $\mathcal{T}(\mathtt{v}) = s \to t$, for example, and require that the semantic annotation of any feature $\mathtt{f}$, and of any feature $\mathtt{\approx f}$, be of type $\mathcal{T}(\mathtt{f})$.

Returning now to the schema for MERGE in (19), the semantic value $\alpha \& \theta_1(\beta)$ must be of type $\mathcal{T}(\mathtt{g})$, as must therefore $\theta_1(\beta)$ itself. Since β must be of type $\mathcal{T}(\mathtt{f}_1)$, θ_1 must be of type $\mathcal{T}(\mathtt{f}_1) \to \mathcal{T}(\mathtt{g})$. The semantic annotation of a selecting feature, then, must be a function mapping the semantic type corresponding to the selected syntactic category, to the semantic type corresponding to the syntactic category at the end of its feature sequence. Setting $\mathcal{T}(\mathtt{d}) = e$, the $\mathsf{Stabbee}$ annotation of the $\mathtt{=d}$ feature of ‘stabbed’, for example, is of type $\mathcal{T}(\mathtt{d}) \to \mathcal{T}(\mathtt{v})$ or $e \to (s \to t)$, a function from individuals to event predicates. This can be thought of as an implementation of Carlson’s [14] idea that thematic roles adjust NP denotations to make them suitable for intersection (conjunction) with a set (predicate) of events.

This theory accounts for the correlation of properties of Class 1 words and Class 2 words observed in (2-6). Class 1 words are those that have as semantic value a predicate with type $\mathcal{T}(\mathtt{v})$, ready for conjunctive interpretation; they are therefore optional but can be added without bound. Class 2 words are those that have a semantic value of some other type, and must therefore interact with a thematic relation for interpretation; since each verb makes available a precise finite number of these thematic relations, a precise finite number of these words will need to appear.

6 Generalising Beyond the Verbal Domain

If these schemas in (19) and (20) were to be adopted for the MERGE and ADJOIN operations throughout the grammar, a problem would arise. The arbitrary category $\mathtt{f}$ in the ADJOIN schema must correspond some monadic predicate type, in order for the schema to be well-typed; but the nominal expressions of category `d` that are selected by verbs, which I have assumed denote individuals (type e), are thought to be possible adjunction sites as well. Indeed, adjunction to *any* category is generally thought to be possible, suggesting that for any category `f`, $\mathcal{T}(\mathtt{f}) = \tau \rightarrow t$ for some semantic type τ.

Pietroski [13, 19] has suggested exactly this: that *every* expression of natural language denotes some monadic predicate. Space constraints preclude rehearsing the independent empirical justification for this here, but the proposal allows for the distinction noted in the verbal domain, between the two ways to modify a neo-Davidsonian event variable, to be extended to other categories. A canonical case of adjunction in the nominal domain, with a clear conjunctive interpretation, is the attachment of a relative clause. This is illustrated in the following application of the ADJOIN schema, where **tall** and **b** are both *predicates of individuals* (type $e \rightarrow t$):

$$\frac{\text{Brutus} :: \langle \mathtt{d}, \mathbf{b}\rangle \qquad \text{who is tall} : \left\langle \mathtt{\approx d}, \mathbf{tall}\right\rangle}{\text{Brutus who is tall} : \langle \mathtt{d}, \mathbf{b} \mathbin{\&} \mathbf{tall}\rangle}\ \text{\textsc{adjoin}}$$

The interpretation of the checking of selecting features, upon application of MERGE, must now involve existential closure of the predicate denoted by the selected expression. We define in (21) a unary operator $\cdot^{\exists}$ that modifies the annotations of selecting features (i.e. generalisations of thematic roles) to include the appropriate existential closure, such that the truth condition in (22b) can be written as in (22c).

(21) $$R^{\exists}(P) := \lambda e.\exists x[P(x) \wedge R(x)(e)]$$

(22) a. Brutus, who is tall, stabbed Caesar

b. $\exists e[\mathbf{stabbing}(e) \wedge \exists x[(\mathbf{b} \mathbin{\&} \mathbf{tall})(x) \wedge \mathsf{Stabber}(x)(e)] \wedge \exists y[\mathbf{c}(y) \wedge \mathsf{Stabbee}(y)(e)]]$

c. $\exists e[(\mathbf{stabbing} \mathbin{\&} \mathsf{Stabber}^{\exists}(\mathbf{b} \mathbin{\&} \mathbf{tall}) \mathbin{\&} \mathsf{Stabbee}^{\exists}(\mathbf{c}))(e)]$

So if R is (the curried characteristic function of) a binary relation between elements of τ_1 and τ_2, and P is a predicate of elements of τ_1, then $R^{\exists}(P)$ is the predicate satisfied by $e \in \tau_2$ iff there is some $x \in \tau_1$ satisfying P such that $R(x)(e)$. For example, $\mathsf{Stabber}^{\exists}(P)$ is a predicate satisfied by those events of which a stabber satisfies P.

The revised schema for the MERGE operation in (23) applies this $\cdot^{\exists}$ operator to the annotation of the selecting features.

(23) $$\frac{s_1 :: \langle \mathtt{=f}_1, \theta_1\rangle \ldots \langle \mathtt{=f}_n, \theta_n\rangle \, \langle \mathtt{g}, \alpha\rangle \qquad s_2 :: \langle \mathtt{f}_1, \beta\rangle}{s_1 s_2 : \langle \mathtt{=f}_2, \theta_2\rangle \ldots \langle \mathtt{=f}_n, \theta_n\rangle \left\langle \mathtt{g}, \alpha \mathbin{\&} \theta_1^{\exists}(\beta)\right\rangle}\ \text{\textsc{merge}}$$

Now the semantic annotation θ_i for each selecting feature $\mathtt{=f}_i$ is of type $\mathrm{dom}(\mathcal{T}(\mathtt{f}_i)) \rightarrow \mathcal{T}(\mathtt{g})$, such that $\theta_i^{\exists}$ is of type $\mathcal{T}(\mathtt{f}_i) \rightarrow \mathcal{T}(\mathtt{g})$. Then the last step in the derivation of (22a) will proceed as shown in Fig. 2(c).

The complete lexicon for the examples I have considered is given in (24).

(24)
$$\begin{array}{rlrl}
\multicolumn{4}{c}{\text{stabbed} :: \langle \mathtt{=d}, \mathsf{Stabbee} \rangle, \langle \mathtt{=d}, \mathsf{Stabber} \rangle, \langle \mathtt{v}, \mathbf{stabbing} \rangle} \\
\text{Brutus} :: & \langle \mathtt{d}, \mathbf{b} \rangle & \text{yesterday} :: & \left\langle \mathtt{\approx v}, \mathbf{yesterday} \right\rangle \\
\text{Caesar} :: & \langle \mathtt{d}, \mathbf{c} \rangle & \text{violently} :: & \left\langle \mathtt{\approx v}, \mathbf{violent} \right\rangle \\
\text{who is tall} :: & \left\langle \mathtt{\approx d}, \mathbf{tall} \right\rangle & & \\
\text{where:} & \multicolumn{3}{r}{\mathbf{stabbing}, \mathbf{yesterday}, \mathbf{violent} \in (s \to t)} \\
& \multicolumn{3}{r}{\mathbf{b}, \mathbf{c}, \mathbf{tall} \in (e \to t)} \\
& \multicolumn{3}{r}{\mathsf{Stabber}, \mathsf{Stabbee} \in (e \to (s \to t))}
\end{array}$$

If we limit our attention to applications of MERGE, then syntactic types (i.e. feature sequences) uniquely determine semantic types, just as in familiar categorial grammars. If in addition we assume that no semantic type corresponds to more than one syntactic category (i.e. that $\mathcal{T}$ is injective), there is a one-to-one correspondence between syntactic types and semantic types: whereas (most) categorial grammars use distinct slashes to produce two syntactic types for each semantic type as shown in (25), MGs do not distinguish between selection on the left or on the right (see appendix) and therefore we have the single rule in (26).

(25) $$\mathcal{T}(A/B) = \mathcal{T}(B) \to \mathcal{T}(A) \qquad\qquad \mathcal{T}(B\backslash A) = \mathcal{T}(B) \to \mathcal{T}(A)$$

(26) $$\begin{aligned} \mathcal{T}(\mathtt{=f}_1 \ldots \mathtt{=f}_n\ \mathtt{g}) = (\mathrm{dom}(\mathcal{T}(\mathtt{f}_1)) \to \mathcal{T}(\mathtt{g})) \times \ldots \\ \times (\mathrm{dom}(\mathcal{T}(\mathtt{f}_n)) \to \mathcal{T}(\mathtt{g})) \times \mathcal{T}(\mathtt{g}) \end{aligned}$$

With this in mind, we can understand syntactic composition to be "driven by" semantic types when the MERGE rule is applied in (27), much as we can think of the composition in (28) as driven by (simpler and more familiar) semantic types.

(27) $$\frac{\text{Brutus} :: \mathbf{b}_{e\to t} \qquad \text{walked} :: \langle \mathsf{Walker}, \mathbf{walking} \rangle_{(e\to(s\to t))\times(s\to t)}}{\text{Brutus walked} : \mathbf{walking}\ \&\ \mathsf{Walker}^{\exists}(\mathbf{b})}\ \textsc{merge}$$

(28) $$\frac{\text{Brutus} :: \mathbf{b}_{e} \qquad \text{walked} :: \mathbf{walked}_{e\to t}}{\text{Brutus walked} :: \mathbf{walked}(\mathbf{b})}$$

7 Conclusion

This unified system presented in sections 5 and 6 illustrates that the stipulated differences between the two modes of syntactic composition in the MG formalism can be thought of as a transparent reflection of neo-Davidsonian semantic composition. The correlation of distributional properties and semantic properties observed in section 1 is then explained. An expression will either denote a predicate of the same type as the one it attaches to, in which case its interpretation will be strictly conjunctive and it will be able to attach unassisted; or it will denote a predicate of some different type, in which case it will need to interact with an appropriate "thematic" relation, of which the attachment site makes a precise finite number available, for interpretation to be possible.

A Appendix: (A Tiny Subset of) Minimalist Grammars

The aim of this appendix is to make explicit the way in which the MERGE and ADJOIN operations compose the string components of MG expressions, which I have glossed over in this paper. It is certainly not a general introduction to the MG formalism; see [2, 20] for that purpose.

We assume an alphabet Σ and a set $\mathbb{C}$ of categories. The set of features is

$$\mathbb{F} = \mathbb{C} \cup \{\mathsf{=c} : \mathsf{c} \in \mathbb{C}\} \cup \{\mathsf{\approx c} : \mathsf{c} \in \mathbb{C}\}$$

and the set of expressions is

$$\mathbb{E} = \Sigma^* \times \{:, ::\} \times \mathbb{F}^*$$

Lexical expressions or lexical items are expressions with :: as their second coordinate; expressions with : as their second coordinate are non-lexical or complex.

In the original presentation of MGs [2], the non-lexical expressions are binary trees with lexical items at the leaves. These trees allow movement operations to be defined which rearrange the internal structure of a tree, but since I have made no use of these movement operations in this paper we can ignore this additional complexity and instead adopt representations more like, but even simpler than, those of [20].

A language is defined as the closure of a set of lexical items under the structure-building functions: for our purposes, only two are relevant, MERGE and ADJOIN, where MERGE is the union of the two functions MERGE_1 and MERGE_2, defined below. In these definitions c ranges over categories, α ranges over sequences of features, $\cdot$ ranges over $\{:, ::\}$ and s, t range over strings.

$$\frac{s :: \mathsf{=c}\alpha \quad t \cdot \mathsf{c}}{st : \alpha}\ \text{MERGE}_1 \qquad\qquad \frac{t \cdot \mathsf{c} \quad s : \mathsf{=c}\alpha}{ts : \alpha}\ \text{MERGE}_2$$

$$\frac{s \cdot \mathsf{c}\alpha \quad t \cdot \mathsf{\approx c}}{st : \mathsf{c}\alpha}\ \text{ADJOIN}$$

In brief, when a lexical expression "selects" another expression, the string yield of the selectee (a complement) is concatenated on the right (MERGE_1), and when a complex expression selects another expression, the string yield of the selectee (a specifier) is concatenated on the left (MERGE_2); and when an expression adjoins to another, its string yield is concatenated on the right.

In the main text of the paper I take these rules as operations on categorised strings as a starting point, and propose a way to add semantics to them.

References

1. Frey, W., Gärtner, H.M.: On the treatment of scrambling and adjunction in minimalist grammars. In Jäger, G., Monachesi, P., Penn, G., Wintner, S., eds.: Proceedings of Formal Grammar 2002. (2002) 41–52
2. Stabler, E.P.: Derivational minimalism. In Retoré, C., ed.: Logical Aspects of Computational Linguistics. Volume 1328 of Lecture Notes in Computer Science., Springer (1997) 68–95
3. Amblard, M.: Répresentations sémantiques pour les grammaires minimalistes. Technical Report 5360, INRIA (2004)
4. Lecomte, A., Retoré, C.: Extending lambek grammars: a logical account of minimalist grammars. In: ACL '01: Proceedings of the 39th Annual Meeting of Association for Computational Linguistics, Morristown, NJ, Association for Computational Linguistics (2001) 362–369
5. Vermaat, W.: Controlling movement: Minimalism in a deductive perspective. Master's thesis, Utrecht University (1999)
6. Kobele, G.M.: Generating Copies: An investigation into Structural Identity in Language and Grammar. PhD thesis, UCLA (2006)
7. Kobele, G.M.: Inverse linking in minimalist grammars. Ms. (2009)
8. Davidson, D.: The logical form of action sentences. In Rescher, N., ed.: The Logic of Decision and Action. University of Pittsburgh Press, Pittsburgh (1967)
9. Davidson, D.: Adverbs of action. In Vermazen, B., Hintikka, M., eds.: Essays on Davidson: Actions and Events. Clarendon Press, Oxford (1985)
10. Castañeda, H.N.: Comments. In Rescher, N., ed.: The Logic of Decision and Action. University of Pittsburgh Press, Pittsburgh (1967)
11. Parsons, T.: Events in the semantics of English. MIT Press, Cambridge, MA (1990)
12. Schein, B.: Plurals and Events. MIT Press, Cambridge, MA (1993)
13. Pietroski, P.M.: Events and Semantic Architecture. Oxford, New York, NY (2005)
14. Carlson, G.: Thematic roles and their role in semantic interpretation. Linguistics **22** (1984) 259–279
15. Heim, I., Kratzer, A.: Semantics in Generative Grammar. Blackwell, Oxford (1998)
16. Larson, R.K.: Events and modification in nominals. In: Proceedings of SALT VIII, Ithaca, NY, CLC Publications (1998) 145–168
17. McNally, L., Boleda, G.: Relational adjectives as properties of kinds. Empirical Issues in Formal Syntax and Semantics **5** (2004) 179–196
18. Schein, B.: Events and the semantic content of thematic relations. In Preyer, G., Peter, G., eds.: Logical Form and Language. Oxford University Press, Oxford (2002) 263–344
19. Pietroski, P.M.: Interpreting concatenation and concatenates. Philosophical Issues **16**(1) (2006) 221–245
20. Stabler, E.P., Keenan, E.L.: Structural similarity within and among languages. Theoretical Computer Science **293** (2003) 345–363