Polynomial Event Semantics
Negation

Oleg Kiselyov[0000−0002−2570−2186]

Tohoku University, Japan
oleg@okmij.org

Abstract. Polynomial event semantics is an interpretation of Neo-Davidsonian semantics in which the thorny event quantification problem does not even arise. Denotations are constructed strictly compositionally, from lexical entries up, and quantifiers are analyzed in situ. All advantages of event semantics, in particular, regarding entailment, are preserved. The previous work has dealt only with positive polarity phrases involving universal, existential and counting quantification.
We now extend the polynomial event semantics to sentences with negation and negative quantification, including adverbial quantification, with attendant ambiguities. The analysis remains compositional, and does not require positing of non-existing entities or events.

1 Introduction

Quantification in (Neo-) Davidsonian event semantics has been the subject of many debates; we remind the so-called 'event quantification problem' in §2 and review the proposed resolutions, or postulates, in §6. The problem becomes especially acute with negation.

We propose an interpretation of Neo-Davidsonian semantics in which the event quantification problem does not even arise. The previous work has [5] laid the foundation and described the compositional but non-Montagovian treatment of universal, existential and counting quantification. Denotations are constructed strictly compositionally, from lexical entries up, and quantifiers are analyzed in situ, with no need for lifting. The underlying machinery is not of lambda-calculus but of much simpler relational algebra, with the straightforward set-theoretic interpretation.

The first key idea, strongly reminiscent of the BHK interpretation of intuitionistic logic [4], is viewing the truth value of a sentence not as the simple true/false but as a set of *evidence* for it: e.g., transpired events that witness for the sentence. Entailment is decided by set inclusion. The denotation then is a *query*, of a database of events. The query, expressed in a relational algebra, is constructed following the structure of the sentence, i.e., compositionally. One query entails another if the result of the former is contained in the result of the latter, for any event database. Queries have no event variable (or any variables for that matter); therefore, the problem of the scope of event quantification does not arise.

The present paper extends the approach of [5] to sentences with negation and negative quantification, including adverbial quantification, with attendant ambiguities. The analysis remains compositional, and does not require positing of non-existing entities or events. The key extension is viewing the truth value of a sentence as a set of evidence as well as *counter-evidence*. The denotation now is a query both for the supporting and the contradicting evidence.

The structure of the paper is as follows. We remind the event quantification problem in §2 and the polynomial event semantics in §3, detailing the treatment of existential quantification in §3.2 and the corresponding entailments in §3.3. Negation is dealt with in §4; in particular, the ambiguity in the presence of quantified adverbial modifiers is analyzed in §4.1. Double-negation is briefly described in §4.2. Section 5 presents a set-theoretical model of the polyconcept algebra. Section 6 discusses related work.

The presented approach is not just a pen-and-paper analysis: it has been implemented so to analyze sentences mechanically and compute their models and counter-models. The implementation, which includes all the example in the paper plus many more, is available at `http://okmij.org/gengo/poly-event/poly.ml`.

2 Event Quantification Problem

We start by recalling the event quantification problem and its particular acute case of negation.

Neo-Davidsonian event semantics [9] (see [8] for a survey) is attractive because of the uniform treatment of VP adverbials, among other things, which explains entailments among sentences without ad hoc meaning postulates. To take the canonical example,

(1) Brutus stabbed Caesar

(2) Brutus stabbed Caesar violently

are given in the Neo-Davidsonian semantics the following denotations (logical formulas), resp.

(3) $$\exists e.\, \mathsf{stabbed}(e) \wedge \mathsf{th}(e){=}\ \mathsf{caesar} \wedge \mathsf{ag}(e){=}\ \mathsf{brutus}$$

(4) $$\exists e.\, \mathsf{stabbed}(e) \wedge \mathsf{th}(e){=}\ \mathsf{caesar} \wedge \mathsf{ag}(e){=}\ \mathsf{brutus} \wedge \mathsf{violent}(e)$$

Here $\mathsf{stabbed}(e)$ and $\mathsf{violent}(e)$ are predicates on the event e (telling if e is a stabbing or violent event, resp); th and ag are thematic functions, which return the theme (resp., agent) for their event argument. Characteristically for the (Neo-) Davidsonian semantics, the event variable e is bound by the existential quantifier at the sentence level. This so-called existential closure lets us interpret the sentence as the proposition: whether an event with the described properties exists. Clearly (4) entails (3) by first-order logic, thus reproducing the entailment from (2) to (1).

There is already a problem when substituting 'every senator' for Caesar:

(5) Brutus stabbed every senator

(6) $\exists e.\, \forall x.\, \mathsf{senator}(x) \implies \mathsf{stabbed}(e) \wedge \mathsf{th}(e){=}x \wedge \mathsf{ag}(e){=}\,\mathsf{brutus}$

whose denotation (6) asserts the existence of a single event of the stabbing spree – 'collective', so to speak, reading of the universal quantifier. This denotation hence cannot reproduce the reading of (5) with stabbing spread over time.

The biggest problem however comes from substituting 'no senator' for Caesar in (1)-(2):

(7) Brutus stabbed no senator

(8) Brutus stabbed no senator violently

If we keep applying the existential closure at the sentence level as before we obtain the following logical formula for (7):

(9) $\exists e.\, \neg\exists x.\, \mathsf{senator}(x) \wedge \mathsf{stabbed}(e) \wedge \mathsf{th}(e){=}x \wedge \mathsf{ag}(e){=}\,\mathsf{brutus}$

which is true if there is any event other than Brutus stabbing some senator (that is, for almost any event). Although the sentences (1) and (7) are contradictory,[1] the denotations (3) and (9) are not: both denotations are true of ancient Rome, for example.

The two problems are instances of what is called the event quantification problem [2, 3]: the problem of scoping of the existential closure with respect to other quantificational phrases, which arises when combining Montagovian semantics and event semantics. Landman [7] suggested so-called scope domain principle, that the existential quantifier for the event variable obligatory takes the lowest scope. The implementations of this postulate are reviewed in §6.

[1] keeping in mind that Caesar was a senator.

We avoid the problem altogether. The story about quantifiers such as 'everyone' was told in [5]. Here we deal with negative-polarity sentences like (7), and also

Brutus did not stab Caesar

Brutus never stabbed Caesar

Brutus did not accuse Caesar for one hour

It is not the case that Brutus never stabbed a senator

3 Polynomial Event Semantics

We now recall the polynomial event semantics from [5], but present it algebraically. Our running example is the earlier (1)-(2), repeated below:

(1) Brutus stabbed Caesar

(2) Brutus stabbed Caesar violently

Suppose we have a record – a database – of events of ancient times. To see if (1) is true, we would query the database for the events of stabbing whose agent/subject is Brutus and theme is Caesar. In the language of set theory, this query may be written as

$$\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{caesar} \tag{10}$$

Here, $\mathsf{Stabbed}$ is the set of stabbing events. As in description logic [1], we call a set of events or individuals a *concept* and typeset in san-serif and capitalized. If subj' is a binary relation (thematic role) between events and their subjects, $\mathsf{subj}'/\mathsf{brutus}$ are those events that are related by subj' to the individual brutus:

$$\mathsf{subj}'/\mathsf{brutus} = \{e \mid \mathsf{subj}'(e, \mathsf{brutus})\} = \{e \mid \mathsf{ag}(e) = \mathsf{brutus}\}$$

$\mathsf{ob1}'/\mathsf{caesar}$ is similar.

Query (10) gives a set of events: each event in this set can act as an *evidence* that Brutus indeed stabbed Caesar – in other words, as a witness for the proposition of (1). This evidence set, or 'support set', may then be regarded as the truth value for the sentence – and the query itself as the denotation.

For (2), the query is

$$\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{caesar} \cap \mathsf{Violently} \tag{11}$$

where $\mathsf{Violently}$ is the set of violent events. Clearly, (11) is a subset of (10), from the very meaning of set intersection. Therefore, if the former is non-empty so is the latter, establishing the entailment from (2) to (1).

3.1 Polyconcepts

In the polynomial event semantics [5] queries are actually written in a more general form, to accommodate quantification. Instead of concepts we will be dealing with polyconcepts – which are also sets, but with more structure (see §5). A concept can be turned into a polyconcept by an injective operator $\mathcal{P}$. The empty polyconcept is written $\perp$, and the polyconcept intersection (a symmetric associative operation) is denoted by $\sqcap$. These operations have the following properties

(12) $$\mathcal{P}(c_1 \cap c_2) = \mathcal{P}c_1 \sqcap \mathcal{P}c_2 \qquad\qquad \mathcal{P}c = \perp \text{ iff } c = \varnothing$$

where the meta-variable c stands for an arbitrary concept. Using polyconcepts, the queries (10) and (11) are written as

(13) $$\begin{aligned}&\mathcal{P}(\mathsf{subj}'/\mathsf{brutus}) \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathcal{P}(\mathsf{ob1}'/\mathsf{caesar})\\ &\quad = \mathcal{P}(\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{caesar})\end{aligned}$$

(14) $$\begin{aligned}&\mathcal{P}(\mathsf{subj}'/\mathsf{brutus}) \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathcal{P}(\mathsf{ob1}'/\mathsf{caesar}) \sqcap \mathcal{P}\,\mathsf{Violently}\\ &\quad = \mathcal{P}(\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{caesar} \cap \mathsf{Violently})\end{aligned}$$

where the equated expressions in (13) and (14) are obtained using (12). Property (12) also lets us conclude that if (14) is non-empty then so must be (13), justifying the entailment. The polyconcept queries are hence an equivalent, but more complicated way of writing the earlier set-theoretic queries. The need for polyconcept comes when we turn to quantification.

We should stress the overt absence of the existential closure. To decide entailments, which is one of the main goals of semantics, working with 'support sets' as they are – or the queries that symbolize them – is enough. The queries are expressed in the form of a relational algebra (description logic [1], to be precise) and have no variables; in particular, no event variable.

Finally, we stress that queries (13) and (14) (as (10) and (11)) look quite like the original sentences (1) and (2). Therefore, the queries (denotations) can be systematically, compositionally constructed from the (parsed tree) of a sentence. As we shall see, this property still holds in the presence of quantification.

3.2 Quantification

The paper [5] extends the query-based semantics to sentences with quantified phrases such as the earlier (5) as well as the following:

(15) Brutus stabbed a senator

We systematically apply the principle that the truth value of a sentence is the set of witnesses for it. A witness for (15) would be a stabbing event with Brutus as the agent and any senator as the theme. A query to search for these events would be a generalization of (13) – or, one may say, a relaxation of (13), where the theme of stabbing events is not just Caesar but any senator:[2]

$$\mathcal{P}\ \mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\ \mathsf{Stabbed} \sqcap \mathcal{P}\ \mathsf{ob1}'/\mathsf{Senator} \tag{16}$$

Here we have extended the rel'/x notation to x being not an individual but a set of individuals (i.e., a concept):

$$\mathsf{ob1}'/\mathsf{Senator} = \{e \mid \mathsf{ob1}'(e, i), i \in \mathsf{Senator}\} = \{e \mid \mathsf{th}(e) \in \mathsf{Senator}\}$$

There is another way to look for the evidence of Brutus' stabbing a senator: query the events database to see if Brutus stabbed Caesar or if Brutus stabbed Antonius, or Cicero, etc. Such 'union' query can be written as

$$\begin{aligned} &(\mathcal{P}\ \mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\ \mathsf{Stabbed} \sqcap \mathcal{P}\ \mathsf{ob1}'/\mathsf{Caesar}) \oplus \\ &\quad(\mathcal{P}\ \mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\ \mathsf{Stabbed} \sqcap \mathcal{P}\ \mathsf{ob1}'/\mathsf{Antonius}) \oplus \ldots \\ &= \bigoplus\nolimits_{i \in Senator} \mathcal{P}\ \mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\ \mathsf{Stabbed} \sqcap \mathcal{P}\ \mathsf{ob1}'/i \end{aligned} \tag{17}$$

if we introduce the $\oplus$ operation to build a 'union' polyconcept out of polyconcept alternatives (pace alternative semantics [11]). One may feel that (16) and (17) ought to be equivalent: indeed, an event of Brutus stabbing a senator would be found by either query. However, as was observed in [5], if the sentence contained a universal quantifier, e.g., 'Every guard stabbed a senator', the corresponding two queries would no longer give the same result. Hence we need to consider both ways to query for the existential evidence.

The operation $\oplus$ is meant to feel like set-union. We likewise regard it as associative and commutative, and let $\sqcap$ distribute similarly to set-intersection distributing through set-union:

$$(x_1 \oplus x_2) \sqcap y = (x_1 \sqcap y) \oplus (x_2 \sqcap y) \qquad\qquad x \oplus x = x \tag{18}$$

Then (17) may be written simpler as

$$\mathcal{P}\ \mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\ \mathsf{Stabbed} \sqcap (\bigoplus\nolimits_{i \in Senator} \mathcal{P}\ \mathsf{ob1}'/i)$$

[2] We often drop the parentheses in $\mathcal{P}(\mathsf{subj}'/\mathsf{brutus})$, etc. if no confusion results.

To simplify the notation even further, we introduce the operator $\mathcal{A}$ turning a concept into a polyconcept:

$$\mathcal{A}c = \bigoplus\nolimits_{i \in c} \mathcal{P}\{i\}$$

Unlike $\mathcal{P}c$, the polyconcept $\mathcal{A}c$ treats each element of c as its own alternative. Extending the notation rel'/x one more time, to x being a polyconcept:

$$\mathsf{rel}'/(\mathcal{P}\mathsf{c}) = \mathcal{P}\,\mathsf{rel}'/c \qquad\qquad \mathsf{rel}'/(\mathcal{A}\mathsf{c}) = \bigoplus\nolimits_{i \in c} \mathsf{rel}'/i$$

lets us finally write the two queries expressing the meaning of (15) as:

(19) $\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\mathcal{P}\mathsf{Senator}$

(20) $\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\mathcal{A}\mathsf{Senator}$

One may have noticed that (17) did not look compositionally constructed. On the other hand, queries (19) and (20) both clearly match the structure of sentence (15) and are constructed compositionally. One may then conclude that $\mathcal{P}\,\mathsf{Senator}$ and $\mathcal{A}\,\mathsf{Senator}$ are two ways to denote 'a senator'. (If we had more quantifiers, we would have observed that the former is the narrow-scope existential and the latter is wide-scope: see [5] for discussion.) In polynomial semantics, existentials (and other quantifiers, for that matter) are analyzed in situ, with no movements.

3.3 Existential Quantification and Entailment

Deciding entailment in the polynomial event semantics is hardly any different from the ordinary Neo-Davidsonian semantics, even in the presence of (existential) quantification. For example, consider (15) (repeated below) and (21)

(15) Brutus stabbed a senator

(21) Brutus stabbed a senator violently

Similarly to (19) and (20), the meaning of (21) is expressed by:

(22) $\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\mathcal{P}\mathsf{Senator} \sqcap \mathcal{P}\,\mathsf{Violently}$

(23) $\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\mathcal{A}\mathsf{Senator} \sqcap \mathcal{P}\,\mathsf{Violently}$

In our evidence-based approach, one sentence is said to entail another if any evidence for the former is, or gives, the evidence for the latter, for any event database. More formal, and useful for polyconcepts, definition is that a sentence denoted by the polyconcept x entails another, denoted by y, just in case $y \neq \bot$ whenever $x \neq \bot$ – for any event database. It is easy to see, from (12) and (18) that $x \sqcap y \neq \bot$ always implies $x \neq \bot$. Therefore, (21) entails (15).

With a bit more work one can show that if $(\mathcal{P}c_1 \oplus \mathcal{P}c_2) \sqcap x$ is not $\bot$ then neither is $\mathcal{P}(c_1 \cup c_2) \sqcap x$. That is, that the wide-existential reading, such as (23), entails the narrow-existential reading, such as (22) (in the sentences without negation).

4 Negation

Our principle has been that the truth value of a sentence is a (poly-)set of witnessing events. Applying it to sentences like (7) and (8), repeated below

(7) Brutus stabbed no senator

(8) Brutus stabbed no senator violently

is a challenge: how can one witness something that has not occurred? Our resolution is to consider 'counter-witnesses': events that testify against the sentence. The truth value of a sentence hence becomes a set of witnesses and a set of counter-witnesses (or, refutations). To evaluate (7) we would query the database of events for senator stabbings done by Brutus. The empty result would mean (7) is *non-refuted* by the available evidence.

Formally, we extend the previously introduced polynomial event semantics by assigning polarity: Positive polyconcepts characterize supporting, and negative polyconcepts — refuting events. The empty polyconcepts are also polarized: $\bot$ resp. $\bar{\bot}$, which are distinct. The operations $\mathcal{P}$ and $\mathcal{A}$ create positive polyconcepts. For negative ones we introduce negation $\neg x$, with the property

(24) $\neg x \sqcap y = \neg(x \sqcap y) \quad \neg(x \oplus y) = \neg x \oplus \neg y \quad \mathsf{rel}'/\neg x = \neg\, \mathsf{rel}'/x$

where x and y are assumed of positive polarity. (For double-negation, see §4.2.)

We have seen in §3.2 that 'a senator' may be represented either by $\mathcal{A}\mathsf{Senator}$ or $\mathcal{P}\mathsf{Senator}$. In the former, 'wide-scope' reading, the polyconcept contains $\oplus$-collected alternatives for each particular senator. The 'narrow-scope' reading collapses them. Since 'no senator' does not focus (pun intended) on individual people, it seems reasonable to give it only one interpretation: $\neg\mathcal{P}\,\mathsf{Senator}$. Thus, 'no senator' is adversarially testifying narrow-scope 'a senator'.

The meaning of (7) is hence the query

$$\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\neg\mathcal{P}\mathsf{Senator} \tag{25}$$

$$= \neg\mathcal{P}(\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{Senator}) \tag{26}$$

where (26) is obtained by applying the properties of polyconcept operations. The result, if not $\bar{\bot}$, carries an event of Brutus stabbing a senator: the evidence refuting (7).

For (8) we obtain

$$\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\neg\mathcal{P}\mathsf{Senator} \sqcap \mathcal{P}\,\mathsf{Violently} \tag{27}$$

$$= \neg\mathcal{P}(\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{Senator} \cap \mathsf{Violently}) \tag{28}$$

From (12) and the properties of set-intersection we obtain that if (26) is $\bar{\bot}$, then so must be (28) – meaning that (7) entails (8). In general, one may observe that the operator $\sqcap$ is upwards monotone. Therefore, dropping $(\sqcap\ \mathcal{P}\,\mathsf{Violently})$ does not reduce supporting or refuting evidence – letting us decide entailments such as 'no guard stabbed Caesar' entailing 'no guard stabbed Caesar violently' without any meaning postulates, just by monotonicity of $\sqcap$.

Negated verbs such as 'do not stab' are represented by applying $\neg$ to the verb's concept. (Adverbs like 'never' are treated similarly, as the negated concept 'ever'; we look at time-period–related concepts in §4.1). Thus the meaning of (29) is (30)

(29) Brutus did not stab Caesar

$$\begin{aligned} &\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \neg\mathcal{P}\,\mathsf{Stabbed} \sqcap \mathcal{P}\,\mathsf{ob1}'/\mathsf{caesar} \\ &= \neg\mathcal{P}(\mathsf{subj}'/\mathsf{brutus} \cap \mathsf{Stabbed} \cap \mathsf{ob1}'/\mathsf{caesar}) \end{aligned} \tag{30}$$

as expected. Likewise, for (31) we obtain the query (32)

(31) Brutus did not stab a senator

$$\mathcal{P}\,\mathsf{subj}'/\mathsf{brutus} \sqcap \neg\mathcal{P}\,\mathsf{Stabbed} \sqcap \mathsf{ob1}'/\mathcal{P}\mathsf{Senator} \tag{32}$$

The paper [5] described in detail how ambiguities in sentences like "A soldier stabbed everyone" are reflected in the polynomial event semantics. The just shown treatment of 'do not stab' predicts that "A soldier did not stab everyone" will be just as ambiguous. We deal with ambiguous negative sentences in more detail next.

4.1 Scope ambiguities with quantified adverbial modifiers

It has long been observed (see [2] for references and detailed discussion) that negative sentences with for-adverbials like 'for one hour' are ambiguous. For example,

(33) Brutus did not accuse Caesar for one hour

may be paraphrased either as (34) or (35), 'for one hour' taking scope above or under the negation.

(34) There was an one-hour period during which Brutus did not accuse Caesar

(35) It was not the case that Brutus accused Caesar for one hour

The first comprehensive treatment of this phenomenon in event semantics was done by Krifka [6]. The source of much of the complexity in his very complicated treatment was the desire to avoid having for-adverbials necessarily take the sentence-wide scope (otherwise, overgeneration occurs). Later Champollion [2] delivered a much simpler and compelling analysis in his compositional event semantics, still avoiding the sentence-wide scope of for-adverbials and accounting for tense and sub-interval quantification, as in [6].

Yet another event-based analysis is proposed in [3], using abstract categorial grammar. However, that analysis [3, eq. (36)], makes significant simplifying assumptions: it lets for-adverbials take the sentence-wide scope, and also disregards tense. Also it does not quite convey the meaning of (their version of) (34), which states that there was one hour period during which Brutus did not accuse Caesar, even for a moment. The analysis of [3] however assumed that an accusation action necessarily spans the entire one-hour period.

For-adverbials have the inherently complex semantics, referring not just to an interval of time but also to all sub-intervals of that interval (or all (sub)events that occurred during that interval). Since we eschewed universal quantification in this paper, we will not analyze (34) in all its complexity, assuming, like [3], that the accusation spans the entire period. We do avoid the need for sentence-wide scoping of 'for one hour', and can account for tense (along the lines of [2, 6]). We also exhibit the ambiguity.

We take the concept 'for one hour', denoted as `1hr`, to be a set of events that lasted for one hour, within some reference time frame. We implicitly assume that all queries search for events within the reference time frame determined from tense markers – following the anaphoric treatment of tense in the style of [10], also used in [6] and [2]. The concept $\mathsf{1hr}$ can be turned into a polyconcept in two distinct ways: as $\mathcal{P}\mathsf{1hr}$ or $\mathcal{A}\mathsf{1hr}$. Then (36) is the query representing the meaning of (34), and (37) representing the meaning of (35).

(36) $\mathcal{P}\,\mathsf{subj'/brutus} \sqcap \neg\mathcal{P}\,\mathsf{Accused} \sqcap \mathcal{P}\,\mathsf{ob1'/caesar} \sqcap \mathcal{A}\,\mathsf{1hr}$

(37) $\mathcal{P}\,\mathsf{subj'/brutus} \sqcap \neg\mathcal{P}\,\mathsf{Accused} \sqcap \mathcal{P}\,\mathsf{ob1'/caesar} \sqcap \mathcal{P}\,\mathsf{1hr}$

Indeed, (37) looks for any event of Brutus' accusing Caesar for a one-hour period within the reference time frame, delivering the result as the single alternative. If it is $\bar{\bot}$, then no such events are found and (35) is non-refuted. On the other hand, (36) delivers the refutation events as multiple alternatives, one per each 1hr period. An alternative $\bar{\bot}$, if present, would then non-refute (34).

4.2 Double Negation

Standard English generally does not allow multiple negations, at least overtly. (Although combining negation with verbs like 'deny' is grammatical, the meaning is not easy to grasp. Native speakers are routinely confused: see the extensive 'Archive of Misnegation' maintained by Language Log.[3])

Yet there is the construction "It is not the case that S" in which the clause S may already be negated. In that case, the construction performs the classical double negation. For example:

(38) Brutus never stabbed a senator

(39) It is not the case that Brutus never stabbed a senator

Here, (38) denies but (39) affirms a stabbing.

Our treatment of negation easily explains such behavior. Recall our earlier example:

(15) Brutus stabbed a senator

In §3.2 we derived the polyconcept for its meaning; let us call it $\mathsf{y_{bss}}$. The meaning of (38) then works out to be $\neg\,\mathsf{y_{bss}}$ (similarly to the derivation of (32)). If (39) is deemed to be the negation of (38), its meaning then is represented by $\neg\neg\,\mathsf{y_{bss}}$. If $\mathsf{y_{bss}}$ is not $\bot$ it carries an event of Brutus' stabbing some senator, which supports (15). Then $\neg\,\mathsf{y_{bss}}$ is not $\bar{\bot}$, which

[3] `https://languagelog.ldc.upenn.edu/nll/?cat=273`

means (38) is refuted – by the same event, in fact. The very same event witnesses (39). If, however, $\neg\, \mathsf{y_{bss}}$ is $\bar{\bot}$ (that is, (38) is non-refuted), then $\mathsf{y_{bss}}$ must be empty: there are no events to support (15), nor (39). All in all, we see that the negation of $\neg\, \mathsf{y_{bss}}$ is indeed tantamount to $\mathsf{y_{bss}}$.

5 A Model of Polyconcepts

So far we have used polyconcepts as abstract entities with operations $\mathcal{P}$, $\sqcap$, $\oplus$, $\neg$. We postulated desired properties of these operations, and intuitively justified them by analogy with operations on sets. One cannot help but ask: does such polyconcept algebra really exist? Is there a concrete mathematical structure on which we can define $\oplus$, etc. that actually possess the postulated properties? In other words, is there a model of polyconcepts?

This section exhibits a set-theoretic model. It is based on the model introduced in [5], with one simplification and one extension. Paper [5] dealt with the universal and counting quantification (out of scope for the present paper); omitting it gives a simpler model. The extension is the polarity, to deal with negation.

Following the terminology of [5], we call events, humans and other entities *individuals*, and use the meta-variable i to refer to an individual. We call a possibly empty set of individuals a *concept*, referred to by the meta-variable c. A *factor* is a polarized concept.[4] A positive factor is written just as the corresponding concept, using the meta-variable c. A negative factor is written as $\bar{c}$. A polyconcept then is a set of factors, for which we use the meta-variables x and y.

The operations on polyconcepts are defined as follows.

$$
\begin{array}{lll}
\mathcal{P}c & := \{c\} & = \{\bigcup_{i\in c}\{i\}\} \\
\mathcal{A}c & := \bigoplus_{i\in c}\{\{i\}\} & = \bigcup_{i\in c}\{\{i\}\} \\
\bot & := \{\varnothing\} & \\
\bar{\bot} & := \{\bar{\varnothing}\} & \\
\mathbf{0} & := \varnothing & \\
x \oplus y & := x \cup y & \\
x \sqcap y & := \{c_1 \cap c_2 \mid c_1 \in x, c_2 \in y\} & \\
\neg x & := \{\bar{c} \mid c \in x\} & \\
\mathsf{rel}'/x & := \{\mathsf{rel}'/c \mid c \in x\} &
\end{array}
$$

[4] To witness universal quantification, [5] introduces a so-called group of events. A factor is then a set of groups. We do not deal with the universal or counting quantification in this paper, and so elide groups, and the related operation $\otimes$ for clarity.

$\mathcal{P}c$ is thus a polyconcept made of a single positive factor. In contrast, $\mathcal{A}c$ is a set of positive singleton factors. Clearly, $\perp$ and $\bar{\perp}$ are distinct, and both are different from $\mathbf{0}$. (We have not used $\mathbf{0}$ before: it is the unit of $\oplus$, see below.) The operation $\oplus$ unions the factors of its polyconcept arguments. When computing $x \sqcap y$ and intersecting factors, if one factor is negative the resulting factor is also negative. The intersection of two negative factors is not defined (it can be permitted in languages with negative concord). If rel' is a binary relation, the sectioning notation rel'/x applies to each factor of x (keeping its polarity).

Below is the summary of the properties of the polyconcept operations; most of them have already been mentioned earlier. It is easy to see that the just defined operations do have all these properties.

$$
\begin{array}{ll}
\multicolumn{2}{c}{\oplus, \sqcap \text{ are associative and commutative}} \\
\mathcal{P}(c_1 \cap c_2) & = \mathcal{P}c_1 \sqcap \mathcal{P}c_2 \\
\mathcal{P}c & = \perp \text{ iff } c = \varnothing \\
x \sqcap \mathbf{0} & = \mathbf{0} \\
x \sqcap \perp & = \perp \quad \text{if all factors of } x \text{ are positive} \\
x \sqcap \bar{\perp} & = \bar{\perp} \quad \text{if all factors of } x \text{ are positive} \\
(x_1 \oplus x_2) \sqcap y & = (x_1 \sqcap y) \oplus (x_2 \sqcap y) \\
x \oplus x & = x \\
\neg x \sqcap y & = \neg(x \sqcap y) \quad \text{if all factors of } x, y \text{ are positive} \\
\neg(x \oplus y) & = \neg x \oplus \neg y \quad \text{if all factors of } x, y \text{ are positive} \\
\mathsf{rel}'/\neg x & = \neg\, \mathsf{rel}'/x \quad \text{if all factors of } x \text{ are positive}
\end{array}
$$

6 Related Work

The problems of quantification and negation in event semantics are well-known and well-described; see [2, 3] for the recent detailed discussion. The proposed resolutions all (except for Krifka's [6] unusual and controversial treatment of negation) center around making the existential quantifier that binds the event variable obligatorily take the lowest scope. The postulate of existential closure having the lowest scope is the generalization of the 'scope domain principle' by Landman [7].

The approaches also effect this lowest scope taking in the same way: existential closure is postulated at the sentence (sometimes VP) level, and other scope-taking operators are *moved* over it. The approaches differ in how exactly this movement happens. So-called syntactic approaches (see [7] for an overview) posit this movement by fiat, as a covert movement or other such operation on the parsed form of the sentence. The abstract categorial grammar approaches [3] postulate abstract types in such a way so that the scope taking operators have no choice but to take scope over existential closure in the so-called abstract form of the sentence. Semantic approaches, rather than postulating a movement upfront, postulate type shifting (or, type-lifting), whose result is the same sort of movement but accomplished during normalizing the denotation.

Of these semantic approaches, Champollion's [2] is notable for using the movement also for existential closure. On his account, the existential quantifier that binds the event variable is included in the lexical entry of a verb, and moved into the sentence or VP scope by the continuation-taking/scope-taking mechanism underlying all semantic approaches. Champollion arranges for stacking-up continuations (in other words, for stacking-up type lifting) in such a way so that the existential closure comes always in the lowest scope with respect to other scope-taking operators.

Positing the existential quantifier for an event in a lexical entry for a verb is a rather strong assumption, as Tomita [12] demonstrated in the analysis of infinitival complements. It commits one to the existence of an event even in sentences such as "Mary forbade every student to leave", where no event related to leaving is ever asserted to take place. Tomita [12] proposes non-existing eventualities to deal with this problem. Applying polynomial semantics to perception reports and infinitival complements is the subject of the future work.

Polynomial event semantics was first introduced in [5] (see that paper also for an overview of related work.) That paper thoroughly employed model-theoretic approach, in the explicit set-theoretic notation similar to that in §5. The present paper pursues the algebraic treatment.

7 Conclusions

We described the extension of the polynomial event semantics to deal with negation and negative quantification. We thus demonstrated how upwards and downwards entailments and quantification ambiguities can be analyzed without resorting to existential closure. As befits the event semantics, the entailments involving verbal modification (such as 'violently') come out set-theoretically, from the properties of set intersection, without resorting to any meaning postulates.

The key idea is defining the truth value of a sentence in terms of events that support or refute it. The denotation of a sentence is represented by a query, which searches for supporting and refuting events in a 'world events' database. The sentence meaning hence becomes fine-grain: a sentence may be supported or unsupported, and also refuted or non-refuted.

A sentence like 'Exactly two people came to the party', treated as the conjunction 'At least two but no more than two people came to the party' can be both supported (in part) and refuted (in part) by an event of three people coming. It is the subject of future work to analyze such conjunctions, and coordination in general, as well as modality.

The grouping and distributing events through factors, which underlies our treatment of quantificational phrases, holds the promise for the uniform approach to collective and distributive quantification. That is one of our ultimate goals.

Acknowledgments I am grateful to anonymous reviewers for very helpful comments and suggestions. I thank Daisuke Bekki for insightful and stimulating questions. This work was partially supported by a JSPS KAKENHI Grant Number 17K00091.

Bibliography

[1] Baader, F.: Description logics. In: Reasoning Web: Semantic Technologies for Information Systems, 5th International Summer School 2009, Lecture Notes in Computer Science, vol. 5689, pp. 1–39. Springer–Verlag (2009)

[2] Champollion, L.: The interaction of compositional semantics and event semantics. Linguistics and Philosophy **38**(1), 31–66 (2015)

[3] de Groote, P., Winter, Y.: A type-logical account of quantification in event semantics. In: JSAI-isAI Workshops. Lecture Notes in Computer Science, vol. 9067, pp. 53–65. Springer (2014), `https://hal.inria.fr/hal-01102261`

[4] Iemhoff, R.: Intuitionism in the Philosophy of Mathematics. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, fall 2020 edn. (2020)

[5] Kiselyov, O.: Polynomial event semantics - Non-Montagovian proper treatment of quantifiers. In: JSAI-isAI Workshops. Lecture Notes in Computer Science, vol. 11717, pp. 313–324. Springer (2018). https://doi.org/10.1007/978-3-030-31605-1_23

[6] Krifka, M.: Nominal reference, temporal consitution, and quantification in event semantics. In: Bartsch, R., van Benthem, J., van Emde Boas, P. (eds.) Semantics and Contextual Expression, pp. 75–111. Foris Publications, Dordrecht (1989)

[7] Landman, F.: Events and Plurality. Kluwer Academic Publishers, Dordrecht (2000)

[8] Maienborn, C.: 8. Event semantics, pp. 232–266. De Gruyter Mouton (2019). https://doi.org/10.1515/9783110589245-008

[9] Parsons, T.: Events in the Semantics of English: a Study in Subatomic Semantics. The MIT Press, Cambridge, Massachusetts (1990)

[10] Partee, B.H.: Some structural analogies between tenses and pronouns in English. Journal of Philosophy **70**(18), 601–609 (1973). https://doi.org/10.2307/2025024

[11] Rooth, M.: Alternative Semantics. Oxford University Press (02 2016). https://doi.org/10.1093/oxfordhb/9780199642670.013.19

[12] Tomita, Y.: Event quantification in infinitival complements: A free-logic approach. In: Kojima, K., Sakamoto, M., Mineshima, K., Satoh, K. (eds.) JSAI-isAI Workshops 2018. Lecture Notes in Computer Science, vol. 11717, pp. 372–384. Springer (2019). https://doi.org/10.1007/978-3-030-31605-1_27