

1

The Role of Event-Based Representations and Reasoning in Language

James Pustejovsky

Abstract. This chapter briefly reviews the research conducted on the representation of events, from the perspectives of natural language processing, artificial intelligence (AI), and linguistics. AI approaches to modeling change have traditionally focused on situations and state descriptions. Linguistic approaches start with the description of the propositional content of sentences (or natural language expressions generally). As a result, the focus in the two fields has been on different problems. Namely, linguistic theories try to maintain compositionality in the expressions associated with linguistic units, or what is known as *semantic compositionality*. In AI and in the planning community in particular the focus has been on maintaining compositionality in the way plans are constructed, as well as the correctness of the algorithm that searches and traverses the state space. This can be called *plan compositionality*. I argue that these approaches have common elements that can be drawn on to view event semantics from a unifying perspective, where we can distinguish between the surface events denoted by verbal predicates and what I refer to as the *latent event structure* of a sentence. Latent events within a text refer to the finer-grained subeventual representations of events denoted by verbs or nominal expressions, as well as to hidden events connoted by nouns. By clearly distinguishing between surface and latent event structures of sentences and texts, we move closer to a general computational theory of event structure, one permitting a common vocabulary for events and the relations between them, while enabling reasoning at multiple levels of interpretation.

1.1 Introduction

Reasoning about events and their temporal properties is a capability that all humans possess and a hallmark of intelligent behavior in any cognitive system. Our knowledge of actions and events in the world, as well as our thoughts,

fears, and intentions in our mental lives, are essential for modeling causation, constructing complex plans, hypothesizing possible outcomes of actions, and almost any higher order cognitive task. It is not surprising, therefore, that temporal and event reasoning has been a central area of research in artificial intelligence (AI), linguistics, and natural language processing (NLP) for decades.

Event reasoning requires temporal reasoning, which is itself concerned with representing and reasoning about such anchoring and ordering relationships. This is not surprising, because one of the most difficult problems in reasoning about events is locating them temporally. Consider the following simple discourse illustrating this distinction.

- (1) a. Yesterday, John **fell** while **running**.
 b. He **broke** his leg.

Temporally situating the events in this example involves two independent strategies: establishing a *relative ordering* of the events to each other and a *temporal anchoring* of each event relative to a fixed time, such as the overt temporal expression, *yesterday*, or Reichenbach's speech time (Reichenbach, 1947). For this example, we may wish to temporally anchor the *falling*, *running*, and *breaking* events to the particular time (yesterday) or to the speech time (these are past events), as well as order the events relative to each other; e.g., the running precedes the falling, which precedes the breaking. Reasoning with even such a simple narrative requires identifying at least the three following temporal constraint sets, as shown in (2c).

- (2) a. **fall** \subseteq *yesterday*, **run** \subseteq *yesterday*, **break** \subseteq *yesterday*
 b. **fall** $<$ *speech*, **run** $<$ *speech*, **break** $<$ *speech*
 c. **run** $<$ **fall** $<$ **break**

How these pairwise relations are identified and what constitutes the arguments of these relations, however, has been a major research problem for years, in both AI and linguistics, and is also addressed in detail in Chapter 7.

Using commonsense inference, for example, we would expect the falling event to occur before the breaking event, that the falling terminates the running, and that all three occurred yesterday. On the other hand, reasoning linguistically from the grammatical tense and aspect associated with the verbs, we can only conclude that the events are in the past relative to the speech time, that the falling overlapped the time of the running, and that the event of breaking is unordered relative to the others in the past. Further reasoning

from the lexical aspect of the verbs could suggest that both falling and breaking are not compatible event types with running, but this would be a fairly sophisticated linguistic inference. Perhaps even more significant, these inferences already assume that appropriate events can be defined, identified, and packaged as input to subsequent inference procedures.

In the next section, I review what assumptions these fields have made with respect to the recognition of events and how to model them. We will identify two approaches that have figured prominently in event inferencing, the first from linguistics and the second from AI:

- Verbal predicates and some nominals in language are associated with individual events in the logical representation of a sentence.
- Verbal predicates are associated with fluents occupying begin and end states that are formally linked in a model to indicate change of state.

Which approach one adopts depends to a large extent on what kinds of inferences are being addressed; e.g., sentential semantics, discourse inferences, or reasoning about larger narratives and story understanding. The methodology employed in linguistics, where semantic compositionality is a prominent concern, has generally been a better fit for sentence-based event descriptions, and the approach taken in AI has some distinct advantages for narrative understanding and planning. I then discuss computational resources for event semantics, such as VerbNet, and how data structures reflect the linguistic approach to modeling events. Finally, I present some recent developments within event semantics from generative lexicon (GL), which can be seen as attempts to integrate these two methodologies, where the state-change model from AI is embedded within the compositional model of semantics adopted in the linguistic approach. We will see how this change-of-state (update semantics) model of events can be encoded in VerbNet, thereby providing a computational semantic resource to both the AI and NLP communities.

1.2 Introducing Situations and Events

In the simple narrative in (1b) above the verbs are identified as events, with the temporal orderings shown in (2c). Before we discuss how events are recognized and subsequently ordered, however, it is worth considering alternative strategies for temporally anchoring and ordering the content associated with our utterances, in both discourse and narratives.

There are essentially two major approaches to the problem of temporally anchoring the propositional content denoted by a sentence:

1. TENSE AS MODAL: treat the verbal tense as a temporal modal operator over the propositional content of the sentence; or
2. ARGUMENT REIFICATION: introduce a temporally interpreted argument to the expression denoting the propositional content.

In the TENSE AS MODAL approach, a sentence such as *John was hungry* is treated as a proposition scoped by a modal “past tense” operator, *P* (Prior, 1957; Kamp, 1968), shown in (3a) along with its interpretation.

- (3) a. $\mathbf{P}(\text{hungry}(j))$
 b. $\exists t[t < \mathbf{now} \wedge M, t' \models \text{hungry}(j)]$

The interpretation function for this operator introduces a time, t' , in the model, where $t' < \mathbf{now}$, and the sentence *John is hungry* is true at t' . Further examples of this logic in (4) demonstrate the interpretation of sentences with richer tense and aspect marking.

- (4) a. John will have left Boston.
 $\mathbf{F}(\mathbf{P}(\text{leave}(j, b)))$
 $\exists t_1, t_2[\mathbf{now} < t_1 \wedge \mathbf{now} < t_2 < t_1 \wedge M, t_2 \models \text{leave}(j, b)]$
- b. John was going to leave Boston.
 $\mathbf{P}(\mathbf{F}(\text{leave}(j, b)))$
 $\exists t_1, t_2[t_1 < \mathbf{now} \wedge t_1 < t_2 \wedge M, t_2 \models \text{leave}(j, b)]$
- c. John had already left Boston.
 $\mathbf{P}(\mathbf{P}(\text{leave}(j, b)))$
 $\exists t_1, t_2[t_1 < \mathbf{now} \wedge t_2 < t_1 \wedge M, t_2 \models \text{leave}(j, b)]$

The model theory for a temporal modal logic does not include a domain of individuals denoting events; rather, the satisfaction conditions associated with what we might identify as an event occurring are actually operations over propositions navigating the accessibility relations in world valuations. As a result, it is impossible to refer to events as first-class objects for purposes of coreference and quantification.

Though this approach has been developed into a number of rich and very expressive languages for characterizing computational properties of systems – for example, Linear Temporal Logic (LTL; Pnueli, 1977) and Computation Tree Logic (CTL; Emerson and Sistla, 1984) – for even a simple discourse such as that in (2c), it is difficult to capture the temporal ordering constraints that present themselves, since neither events nor states are treated as individually quantified arguments in the logic.

The ARGUMENT REIFICATION method solves this problem by introducing an additional argument or index in the object language itself, which is used to anchor and identify the evaluation of the propositional content from the sentence in the model. There are two traditions associated with this method, one from the early days of AI and the other coming out of philosophical logic and formal semantics in linguistics. From the first tradition, in order to formalize the situations and actions associated with automated planning, McCarthy (1963) enriched the conventional domain of individuals in first-order logic to include *situations*. A situation is a snapshot of the world, containing the propositions holding at that state. The predicates associated with situations are called *propositional fluents*. Returning to the example in (3), this method reifies the situation encompassing the propositional content of “John being hungry” as s and anchors it to a temporal index, t .

(5) $\exists s, t[\text{hungry}(j, s) \wedge \text{time}(s) = t \wedge t < \text{now}]$

This method was adopted and enriched in McCarthy and Hayes (1969), where it became known as the *situation calculus*, one of the most influential event-based formalisms in logical approaches to AI and formal reasoning. By reifying the propositional content of John’s hunger as a situation variable, it can be referenced for planning algorithms; for example, as one of the preconditions for someone eating food.

Addressing the same problem from the tradition of philosophical logic and language analysis, Davidson (1967) proposed a related but distinct representation for the interpretation of action sentences in natural language. Specifically, he proposed that events (but not states) should be included in the logical description of certain predicates (action verbs) as an additional argument, one that is associated with the verb rather than any overt syntactic argument. Hence, the sentence in (6a) is paraphrased in (6b) and has the logical form of (6c).

- (6) a. Brutus stabbed Caesar.
 b. “There was a stabbing event between Brutus and Caesar.”
 c. $\exists e[\text{stab}(b, c, e)]$

Davidson argued that events are concrete entities that can be perceived, located in space and time, and, moreover, are linguistically real. In (7b), for example, both instances of the pronoun *it* refer to the event “Brutus stabbed Caesar,” and the verb *witness* selects this event as one of its complements.

- (7) a. Brutus **stabbed** Caesar with a knife.
 b. He did **it** and everyone witnessed **it**.

In (7a), the prepositional phrase (PP) *with a knife* can be seen as modifying this event directly, as illustrated in (8).

- (8) a. Brutus stabbed Caesar with a knife.
 b. $\exists e, x[\text{stab}(b, c, e) \wedge \text{knife}(x) \wedge \text{with}(e, x)]$

After the adoption of Davidson's event variable in Parsons (1989), it became a standard method of interpretation for events in formal semantics and eventually in linguistics more broadly (Pustejovsky, 1991; Tenny and Pustejovsky, 2000). In neo-Davidsonian accounts, the event argument does not augment the valency of the verb but is linked to the event participants through semantic roles, such as AGENT (AG), PATIENT (PAT), LOCATION (LOC), and INSTRUMENT (INST), as in (9):

- (9) $\exists e, x[\text{stab}(e) \wedge \text{AG}(e, b) \wedge \text{PAT}(e, c) \wedge \text{knife}(x) \wedge \text{INST}(e, x)]$

Parsons (1990) developed an interpretation of events that introduces a distinction between an event *culminating* (**Cul**) versus an event *holding* (**Hold**). This makes it possible to distinguish the telicity associated with a sentence. Hence, for an event, e , and a temporal interval, t , the following relations hold:

- (10) a. TELIC EVENTS (achievements, accomplishments): **Cul**(e, t)
 b. ATELIC EVENTS (processes, states): **Hold**(e, t)

Returning to the sentence in (8a), we now modify the logical form in (9) to that in (11).

- (11) $\exists e, t, x[\text{stab}(e) \wedge \text{AG}(e, b) \wedge \text{PAT}(e, c) \wedge \text{knife}(x) \wedge \text{INST}(e, x) \wedge \text{Cul}(e, t)]$

Though Davidson's approach appears similar if not identical to the situation calculus, there are significant differences in the way they approach the modeling and representation of events in language (and action or behavior). For Davidsonian models, an action or event in a natural language expression corresponds to an individuated event quantified in the meaning representation associated with that sentence (e.g., (8) and (12b)).

- (12) a. John gave Mary a book.
 b. $\exists e, x[\text{give}(e) \wedge \text{AG}(e, j) \wedge \text{GOAL}(e, m) \wedge \text{TH}(e, x) \wedge \text{book}(x)]$

By reifying the propositional content of the action as a first-order individual in the model, Davidson was able to apply standard first-order logical inference procedures to handle entailments that had previously required higher-order functional behavior. For example, as with the optional arguments to the event in (8) above, both spatial and temporal prepositional phrases modify the event directly, shown in (13b), and not the entire predicate, as seen in (13c).

- (13) a. Brutus stabbed Caesar in the Forum in the morning.
 b. $\exists e[\text{stab}(e) \wedge \text{AG}(e, b) \wedge \text{PAT}(e, c) \wedge \text{LOC}(e, f) \wedge \text{TIME}(e, m)]$
 c. $\text{TIME}(\text{LOC}(\text{stab}(b, c), f), m)$

The template described above, where an atomic event is introduced along with the participants playing specific semantic roles in this event, is also adopted in the representation used in VerbNet, where it provides a computational resource for English verbs, their valencies, and detailed syntactic–semantic descriptions of the constructions associated with each word sense (Kipper et al., 2008).

Now consider how the example in (12ba) is treated within the situation calculus. As mentioned previously, the focus within AI and the planning community was originally on registering and monitoring the state of the world and how properties (fluents) in the world change from one state to the next. Fluents are time-varying properties of individuals. There is no single individual event of “giving” in the situation calculus, because actions are not state descriptions but rather are functions that map states to states and hence act as *state transformers*. The action of giving, for example, has two fluents identifying possession of an object, each associated with a state; i.e., the beginning and end states of the event.

- (14) a. $\exists s_1, t_1[\text{have}(j, x, s_1) \wedge \text{book}(x) \wedge \text{time}(s_1) = t_1]$
 b. $\exists s_2, t_2[\text{have}(m, x, s_2) \wedge \text{time}(s_2) = t_2]$

To model the event in (12b), the initial state, s_1 , is changed by the application of a state-to-state transducer, **give**, which indicates the state that results from its application; i.e., s_2 :

- (15) $\lambda s_2 \exists x, y, z[\text{have}(z, y, \text{Result}(\text{give}(x, y, z), s_2))]$

Though this adequately captures the change in state within the world, where John had the book and now Mary has the book, there are two things missing in this analysis: (a) there is no quantified reference to the change of state itself in the model (i.e., the transition) and (b), there is no event variable associated with any linguistic expression in (12b) denoting this transition. That is, the verbal predicate *give* is not represented at all. This makes the situation calculus difficult to use for modeling natural language in a compositional manner, without some modifications or enrichments.

One such modification was made by McDermott (1982), who abandoned the classical situation calculus notion of events as state transformers. Rather, he defined an event as a *set of intervals* over which a proposition is minimally true (i.e., it happens once): this theory subsequently became an important model for

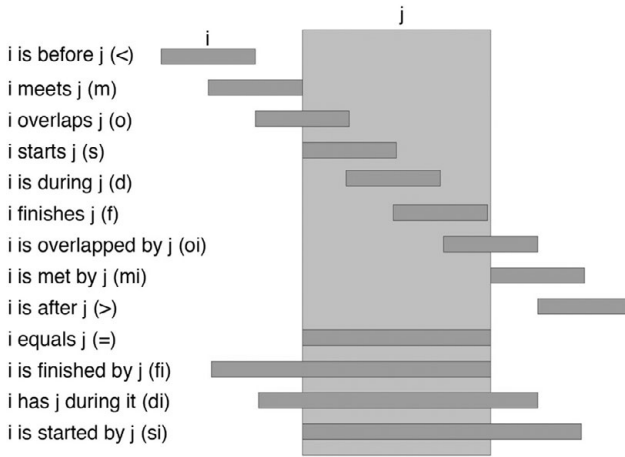


Figure 1.1 Allen's (1984) temporal interval relations.

problem solving and planning paradigms in AI. This is essentially the same intuition that Allen (1984) adopted for his definition of event. A somewhat different approach was taken by Kowalski and Sergot (1989), where events are assumed as primitives in the model, acting as *updates* on the state of the world. In this sense they are *additive* information operations. More specifically, they are seen as actions that initiate or terminate the properties of individuals (the fluents we encountered in the situation calculus).

One of the most influential argument reification methods to be developed within the AI tradition is Allen's interval temporal logic (Allen, 1983). Allen (1984) can be seen as an attempt to model events directly in the logic, while also considering their temporal interval properties (as shown in Figure 1.1) for planning purposes, within general AI concerns. In this system, temporal intervals are considered primitives and constraints (on actions, etc.) are expressed as relations between intervals. There are 13 basic (binary) interval relations, where 6 are inverses of the other, excluding equality.

Allen (1984) made a basic distinction between *properties* (fluents) and *occurrences*, which are similar to Davidson's events and Goldman's actions (Davidson, 1967; Goldman, 1970). As in the situation calculus, a property, *p*, **holds**, if and only if it is true during a specified temporal interval, *t*. This predicate is defined in terms of a downward monotonic subinterval property (Allen, 1984).

$$(16) \text{ holds}(p, t) \leftrightarrow \forall i[in(i, t) \rightarrow \text{holds}(p, i)]$$

Occurrences are a sort of generalized eventuality category that divide into two classes: *processes* and *events*. Processes are characterized as **occurring**, whereas events **occur**. This is essentially identical to Parsons' distinction between **Hold** and **Cul** (Parsons, 1990). As a result, Allen's theory is able to address one of the major issues fundamental to the linguistic approach to event representations, namely, semantic compositionality. By differentiating the satisfaction conditions between the different event types (property, process, and event), Allen's theory can be directly adapted as an interpretation consistent with linguistic treatments of event semantics at the sentence level.

The other major influence in AI promoting the role of events in formal reasoning comes from Hobbs and his colleagues (Hobbs et al., 1987). Hobbs introduced a parsimonious method of treating events as reified first-order individuals in the logic and the model, with an operator he called *nominalization* (Hobbs, 1985). For example, for any predicate, P , in the language, we can generate its nominalized form, P^{Nom} , as follows:

$$(17) \forall x \forall P [P(x) \rightarrow \exists e [P^{\text{Nom}}(e, x) \wedge \text{Exists}(e)]]$$

The *Exists* predicate can be seen as a domain updating operation in the model, ensuring that this individual has the proper extension. As with Allen's interval logic or Davidson's event logic, we can restrict our reasoning to first-order inferences and avoid complications arising from higher-order modifiers and operators. Unlike these approaches, however, Hobbs sees nominalization as a general operation for enriching the ontology, one that can apply to a broad range of conceptual domains, including scales, times, materials, and so on. This general strategy of how ontological types relate to surface realization is given linguistic grounding in Croft (1991), where it is evidenced in language data through a number of typologically broad constructions.

Interestingly, Hobbs' method of reification along with Allen's interval-based notion of events together form the interpretive core of TimeML (Pustejovsky et al., 2003), ISO-TimeML (Pustejovsky et al., 2010), the multilingual resources built on ISO-TimeML community (Im et al., 2009; Bittar et al., 2011; Caselli et al., 2011), as well as the shared tasks based on ISO-TimeML (Verhagen et al., 2007, 2010; UzZaman et al., 2013). ISO-TimeML treats an event as a cover term for any linguistic predicate, as mentioned in a text, that happens or occurs (as with Parsons' and Allen's distinction). A description of the major event categories identified in ISO-TimeML is given below.

(18) a. VERBAL EVENT PREDICATES:

The missile **sank** the ship.

The car **hit** the pedestrian.

- b. NOMINAL EVENTS:
 The **meal** was after the **speech**.
 An **alarm** went off during the **workshop**.
- c. NOMINALIZATIONS:
 The **explosion** occurred at noon.
 The **arrival** of the train was late.
- d. ADJECTIVAL EVENT PREDICATES:
 Mary closed the **open** door.
 They observed the **moving** truck.
- e. PREPOSITIONAL PHRASE PREDICATES:
 John is **on board**.
 Sophie is **in the house**.

As is evident from this list, events can be punctual, last for a period of time, have a logical culmination or no natural termination, or describe states or circumstances, as in the situation calculus. The representation of events as reified intervals with constraints can be mapped to formal calculi used in temporal reasoning, for example DAML-Time (Hobbs and Pustejovsky, 2003), as well as interval temporal logic (Pratt-Hartmann, 2007). This strategy also allows one to interpret the ordering of events in discourse and narratives as an interval constraint satisfaction problem, which has had a significant influence on recognizing narrative event chains and identifying event schemas (Chambers and Jurafsky, 2008, 2009), as well as more recent work on script learning and frame induction (Cheung et al., 2013; Pichotta and Mooney, 2014).

If we interpret Allen's theory directly in terms of a Davidsonian event variable, as with Parsons' treatment discussed above, we can maintain syntactic and semantic compositionality, but we will still lack any representation that reflects the change of state accompanying actions, as with the situation calculus. As it happens, most implementations of Allen's interval calculus are deployed for tracking state-change in planning scenarios or constraint solving problems in NLP involving temporal intervals, rather than for sentence-level linguistic analysis.

What this discussion reveals is that though the underlying formal mechanisms of the situation calculus and event semantics are quite similar, their surface realizations are extremely different. The former focuses on the language of change and state-based situational representations, whereas the latter is concerned with the individuation of events that are denoted by complete sentential expressions. In the next section, we see how these two representational approaches have been integrated into more expressive languages that can

address the core issues in both planning and discourse as well as sentence-level compositionality. We discuss first how to *encode* object properties that change, as subevents in the meaning representations for sentences. We then show how to *track* these object changes over the subevent structures themselves.

1.3 Modeling the Substructure of Events

Consider the philosophical presuppositions associated with the two perspectives discussed above. From an ontological standpoint, the componential structure of *actions* has been taken for granted in AI approaches to modeling change: a CLOSE action brings about a change in state from OPEN(*Y*) to NOT(OPEN(*Y*)). Within linguistics, the componential structure of *words* (not actions or events) has been traditionally of considerable importance, in addition to their various subclasses. To bring these two views closer together, we need to see how words can be associated with the componential structure of actions. But in order to represent the internal structure of events, one must first understand how the meanings of event-denoting expressions are to be represented.

How does this play out? In a logical framework, there are two fundamental ways of representing meaning: (i) the meaning of an accomplishment verb, for example *close*, is the predicate CLOSE, such that a sentence of the form *X closes Y* is mapped to CLOSE(*X*,*Y*); or (ii) the meaning of *close* is decomposed into the meaning of primitive elements; e.g., CAUSE(*X*,BECOME(NOT(OPEN(*Y*))))).

As generative semanticists such as McCawley (1971) and Lakoff (1970) demonstrated, by decomposing the meaning of accomplishment verbs into a causative component along with a result component, we can explain a number of apparent scope ambiguities with adverbs and other modifiers. For example, in (19), *again* can take scope over the entire expression ((19b), the external reading) or the resulting state ((19c), the internal reading).

- (19) a. The astronauts entered the atmosphere again.
 b. AGAIN(ACT(*X*, BECOME(IN(*X*,ATMOSPHERE))))
 c. ACT(*X*, BECOME(AGAIN(IN(*X*,ATMOSPHERE))))

Dowty (1979) developed this decompositional strategy into a fully compositional semantic theory, providing an elegant interpretation for a broad range of linguistic constructions. Interestingly, however, there are still no events in Dowty's theory, not even at the atomic level, because he adopted a classical model with propositional interpretation functions.

Jackendoff (1983), on the other hand, did introduce atomic events into his meaning representations, what he called “conceptual structures.” In Jackendoff (1990) he also assumed that eventualities can be referenced as conceptual categories selected by predicates or functions in the language.

Pustejovsky (1991) can be seen as an attempt to merge the event semantics of Davidson with the structured decomposition of Dowty and Jackendoff, by introducing the notion of subevent structure. In this approach, rather than decomposing an atomic event into predicative primitives, the event itself is decomposed into subevents, each with predicative content (cf. Moens and Steedman, 1988; Pustejovsky, 1988).

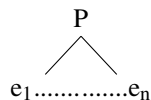
- (20) a. EVENT \rightarrow STATE | PROCESS | TRANSITION
 b. STATE: $\rightarrow e$
 c. PROCESS: $\rightarrow e_1 \dots e_n$
 d. TRANSITION_{ach}: \rightarrow STATE STATE
 e. TRANSITION_{acc}: \rightarrow PROCESS STATE

This is accompanied by meaning-forming rules for event composition, giving rise to the relationships expressed by decomposed meanings. For example, an accomplishment event such as *close* is made up of two aspectually salient parts, a preparatory process with the meaning “ACT(X,Y) \wedge \neg CLOSED(Y)” and a result state with the meaning “CLOSED(Y).” Event structure, in fact, distinguishes the different Aktionsarten in terms of how subevents are structured (cf. (21)), which is a central component of the lexically driven compositionality in GL (Pustejovsky, 1995).

- (21) a. STATE: a simple event, evaluated without referring to other events: *be sick, love, know*

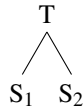


- b. PROCESS: a sequence of events identifying the same semantic expression: *run, push, drag*



- c. TRANSITION: an event identifying a semantic expression evaluated with respect to its opposition: *give, open; build, destroy*

Two-state transition (ACHIEVEMENT): where $\neg\phi \in S_1$, and $\phi \in S_2$



Extended transition (ACCOMPLISHMENT): where $\neg\phi \in P$, and $\phi \in S$



By making explicit reference to the subevents associated with the internal structure of events, this approach enables event representations and the inference systems they support to directly reference and quantify them. Hence, unlike the primitive predicates in Dowty's theory, subevents can be quantified in the logical form of the sentence, in the same way that arguments can be. Consider the inferences associated with the sentences in (22), for example.

(22) a. The destroyer is sinking the boat.

$\exists e_1, x, y[\text{sink_act}(e_1, x, y) \wedge \text{destroyer}(x) \wedge \text{boat}(y)]$

b. The destroyer sank the boat.

$\exists e_1, e_2, x, y[\text{sink_act}(e_1, x, y) \wedge \text{destroyer}(x) \wedge \text{boat}(y) \wedge \text{sink_result}(e_2, y) \wedge e_1 < e_2]$

c. The boat sank.

$\exists e_2, e_1, y, x[\text{sink_result}(e_2, y) \wedge \text{boat}(y) \wedge \text{sink_act}(e_1, x, y) \wedge e_1 < e_2]$

There are distinct advantages to *encoding* state change directly in event structure. Notice that the logical form of the causative (22b) differs from the inchoative (22c) only in the explicit identification of a specific causer. The introduction of an unspecified causer allows the event to be linked in a discourse, where an antecedent or subsequent event might identify the agent. Furthermore, such a representation is essential for recognizing any temporal sequencing that occurs within an event and any changes in state that might have occurred (Pustejovsky, 2005; Mani and Pustejovsky, 2012).

Explicit subevent representations have proved useful, if not crucial, for some types of textual inference, such as *recognizing textual entailment* (RTE; Dagan et al., 2013) and the more recent *natural language inference* (NLI) challenge (Bowman et al., 2015).¹ For example, in order to identify the appropriate discourse relations in a narrative or discourse, it is helpful to

¹ The importance of subevent representation and tracking is discussed in Chapter 3 and the relevance of larger narrative event structures to inference is detailed in Chapters 4 and 6.

know how the events relate to each other semantically and therefore causally. Consider the discourse below.

(23) a. The vase broke.

$\exists e_2, e_1, x, y[\text{broken}(e_2, y) \wedge \text{vase}(y) \wedge \text{break_act}(e_1, x, y) \wedge e_1 < e_2]$

b. John pushed it off the shelf.

$\exists e_3, e_4, z[\text{push_act}(e_3, j, z) \wedge \text{vase}(z) \wedge \text{push_result}(e_4, z) \wedge e_3 < e_4]$

In (23), because of the state resulting from the pushing event, we can conclude that $e_2 = e_4$; further, because $e_1 = e_3$, we have an *explanation* relation between (23b) and (23a) (Hobbs, 1982; Asher and Lascarides, 2003; Im and Pustejovsky, 2010).

The discussion above has focused on the identification and encoding of subevent structure for predicative expressions in language. In subsequent work within GL (Pustejovsky and Moszkowicz, 2011; Mani and Pustejovsky, 2012), event structure is enriched to not only *encode* but dynamically *track* those object attributes modified in the course of the event (the location of the moving entity, the extent of a created or destroyed entity, etc.). The resulting event structure representation is called a *dynamic event structure* (Pustejovsky, 2013). Starting with the view that subevents of a complex event can be modeled as a sequence of states (containing formulae), a dynamic event structure explicitly labels the transitions that move an event from state to state (i.e., programs).²

A dynamic approach to modeling updates makes a distinction between formulae, ϕ , and programs, π . A formula is interpreted as a classical propositional expression, with assignment of a truth value in a specific state in the model (Harel et al., 2000). For our purposes, a state is a set of propositions with assignments to individual variables at a specific frame. We can think of atomic programs as input/output relations, that is relations from states to states, and hence interpreted over an input–output state–state pairing. The model encodes three kinds of representations: (i) predicative **content** of a frame; (ii) **programs** that move from frame to frame; and (iii) **tests** that must be satisfied for a program to apply. These include pretests, while-tests, and result-tests.

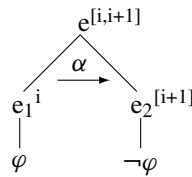
² Each event can be seen as a traced structure over a labeled transition system (van Benthem, 1995). The approach is similar in many respects to that developed in Fernando (2009, 2013) and Naumann (2001).

In this model, there are only two primitive event types: *states*, which are simply propositions describing a snapshot in time, and *transitions*, which are pairs of states connected by a function that moves from the first state to the second state (in some ways similar to the situation calculus representation). These two event types are illustrated in (24).³

(24) a. State



b. Simple Transition

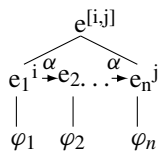


The structure in (24a) represents a **state** as a snapshot of the world in time, e^i , with the propositional content, φ . The event structure in (24b) illustrates how the program α takes the world from the state in e^i with content φ , to the adjacent state, e_2^{i+1} , where the propositional content has been negated, $\neg\varphi$. This structure corresponds directly to **achievements**. From these two types the other two Vendlerian classes can be generated, as we now demonstrate. **Processes** can be modeled as an iteration of simple transitions, where two conditions hold: the transition is a change in the value of an identifiable attribute of the object; every iterated transition shares the same attribute being changed. This is illustrated in (25a).

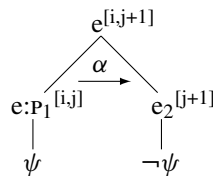
Finally, **accomplishments** are built up by taking an underlying process event, $e:P$, denoting some change in an object's attribute, and synchronizing it with an achievement (simple transition); that is, $e:P$ is unfolding while ψ is true, until one last step of the program α makes it the case that $\neg\psi$ is now true. This can be seen in the event structure in (25b).

(25)

a. Process



b. Accomplishment



³ In Chapter 2, the classical event ontology is enriched substantially to accommodate concepts and relations required for robust processing and interpretation in NLP applications.

1.4 Enriching VerbNet with Event Dynamics

In order to operationalize an event semantics as a computational resource, the information needs to be encoded as frames in the meaning representations associated with the verbal predicates in the language. For modeling predicates with atomic event representations, VerbNet (Kipper et al., 2008) is an example of such an encoding. VerbNet is a lexicon of around 5,200 English verbs, organized primarily around Levin's (1993) verb classification. Classes in VerbNet are structured according to the verb's syntactic behavior, describing the diathesis alternations compatible with each verb (Bonial et al., 2011). Each VerbNet class contains semantic representations expressed as conjunctions of primitive predicates, such as **motion** or **cause** along with an atomic event variable **E**. Event participants that have syntactic relevance are identified with various stages of the event evoked by the syntactic frame. The manner in which a predicate (and its argument) is temporally positioned relative to another is accomplished through the higher-order relation **path_rel**, which requires reference to one of the second-order predicates, **Start**, **During**, or **End**. For example, the semantics for one intransitive frame in class **Run-51.3.2** is shown in (26).

- (26) a. *Billy ran into a cafe.*
 b. **motion**(during(E), Theme)
 path_rel(start(E), Theme, Initial_location, ch_of_loc, prep)
 path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
 path_rel(end(E), Theme, Destination, ch_of_loc, prep)

Hence, what classic VerbNet provides is a rich resource for *atomic* event representations associated with language predicates (i.e., English verbs).

Given our discussion from the previous section, however, there are several things to note about what is not expressed in the VerbNet semantics: (i) the representation is not inherently first-order; (ii) there are no reified subevents that can be referenced in discourse models or planning algorithms; (ii) and there is no mention of the actual change; i.e., the *opposition structure* that is implicated in a change-of-state event, such as *close* or *break*.

Because there is only a single event variable, any ordering or subsetting information needs to be performed as second-order operations. For example, temporal sequencing is indicated with the second-order predicates **start**, **during**, and **end**, which are included as arguments of the appropriate first-order predicates. Attempts to use VerbNet in human–computer interaction concluded that a first-order, enriched subevent representation would greatly facilitate the

interaction between the language parsing and the planning components of the system (Narayan-Chen et al., 2017).

Interpreting (26a) in a first-order atomic representation, where we eliminate the **path_rel** relation and let participant roles act as relations, takes us part of the way, resulting in the event expression given in (27).

$$(27) \exists e, x, l, p [\text{run}(e) \wedge \mathbf{Ag}(e, j) \wedge \mathbf{Init}(e, l) \wedge \mathbf{Dest}(e, x) \wedge \mathbf{Traj}(e, p) \wedge \text{cafe}(x)]$$

As the expression in (27) makes clear, the VerbNet semantic representation includes reference to participants that *can* be realized syntactically but are not always present in a specific construction. A more conventional Davidsonian representation for (26a) would be that shown in (28).

$$(28) \exists e, x [\text{run}(e) \wedge \mathbf{Ag}(e, j) \wedge \mathbf{Dest}(e, x) \wedge \text{cafe}(x)]$$

This leaves the remaining issues unresolved, however. As Zaenen et al. (2008) pointed out, VerbNet is unable to support many temporal and spatial inferencing tasks, because the temporal ordering annotation is not complete or consistent throughout the database; e.g., for several motion classes, **End(E)** was given but not **Start(E)**, and some classes involving change of location of participants (e.g., *gather*, *mix*) did not include a motion predicate at all. That is, from *The diplomat left Bhagdad* you cannot infer *The diplomat was in Bhagdad* (Brown et al., 2018). In addition, attempts to use VerbNet in robotics have resulted in the suggestion that the representation contain more specific causal and temporal relations. The current method of indicating causation, for example, simply had an **Agent** and the event variable *E* as arguments to a **Cause** predicate. This is somewhat misleading in that it could imply that the Agent causes all of *E*, including whatever state exists at **Start(E)**.

A further step toward a proper subeventual meaning representation was proposed in Brown et al. (2018, 2019), where it was argued that, in order to adequately model change, the VerbNet representation must track the change in the assignment of values to attributes as the event unfolds. This includes making explicit any *predicative opposition* denoted by the verb. For example, simple transitions (achievements) encode either an intrinsic predicate opposition (*die* encodes going from $\neg \text{dead}(e_1, x)$ to $\text{dead}(e_2, x)$) or a specified relational opposition (*arrive* encodes going from $\neg \text{loc_at}(e_1, x, y)$ to $\text{loc_at}(e_2, x, y)$). Creation predicates and accomplishments generally also encode predicate oppositions. As we will describe briefly, GL's event structure and its temporal sequencing of subevents solves this problem transparently, while consistent with the idea that the sentence describes a single matrix event, *E*.

With the introduction of GL's event structure, the biggest change to VerbNet is the move from a tripartite division of the temporal span of any event to a model with explicitly quantified (and indexed) subevents, which can be increased or decreased to accommodate the complexity of the event and are ordered using Allen's relational calculus (Allen, 1983). This also eliminates the needs for second-order logic of **Start**(E), **During**(E), and **End**(E), allowing for more nuanced temporal relationships between subevents. The default assumption in this new schema is that e_1 precedes e_2 , which precedes e_3 , and so on. When appropriate, however, more specific predicates can be used to specify other relationships, such as **meets**(e_2, e_3), to show that the end of e_2 meets the beginning of e_3 , or **while**(e_2, e_3) to show that e_2 and e_3 are cotemporal. The latter can be seen in Section 5.1 with the example of accompanied motion.

The second significant change is how causation is represented. Previously in VerbNet, the semantic form implied that the entire atomic event was caused by an Agent; i.e., **cause**(Agent, E). In the revised VerbNet-GL, adopting the GL event structure and assumptions consistent with our discussion above from AI and robotics, "events cause events." Thus, something an agent does (e.g., **do**(e_2 , Agent)) causes a state change or another event (e.g., **motion**(e_3 , Theme)), which would be indicated with **cause**(e_2, e_3). This is seen in (29).

- (29) a. *The lion tamer jumped the lions through the hoop.*
 b. **motion**(during(E), Theme)
 path_rel(start(E), Theme, ?Initial_location, ch_of_loc, prep)
 path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
 path_rel(end(E), Theme, ?Destination, ch_of_loc, prep)
 cause(Agent, E)

To illustrate this change, consider the revised representation in (30) (Brown et al., 2018, 2019).

- (30) *The lion tamer jumped the lion through the hoop.*
 has_location(e_1 , Theme, ?Initial_Location)
 do(e_2 , Agent)
 motion(e_3 , Theme, Trajectory)
 cause(e_2, e_3)
 has_location(e_4 , Theme, ?Destination)

Notice that the predicate **has_location** now has three arguments: an event, e_1 ; a **Theme** argument for the object in motion; and an **Initial_location** argument. The motion predicate is underspecified as to the manner of motion in order to be applicable to the 97 verbs in this VerbNet class. A final **has_location**

predicate indicates the **Destination** of the **Theme** at the end of the event. Any uninstantiated roles in a frame are preceded by ?, such as **Initial_location** and **Trajectory**.

Another way in which VerbNet has accommodated to incorporate an enriched event structure is the encoding of an event-based *opposition structure* to help track the change in object properties over time. This is best illustrated in the revised representation for the VerbNet class **change-of-state**. The representations for changes of state have two basic patterns, depending on whether the change is between absolute states or along a value continuum or scale. The first is illustrated in (31b), the representation for the **Die-42.4** class.

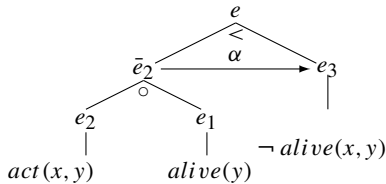
(31) a. *John died.*

b. **cause**(e_2, e_3)

alive($e_1, \text{Patient}$)

¬alive($e_3, \text{Patient}$)

c.



The same event template for relational opposition applies to transfer verbs; e.g., VerbNet class **give-13.1-1**. For example, sentence (32a) has the semantic representation in (32b) and event structure seen in (32c).

(32) a. *Mary gave the book to John.*

b. **transfer_act**(e_2, x, y)

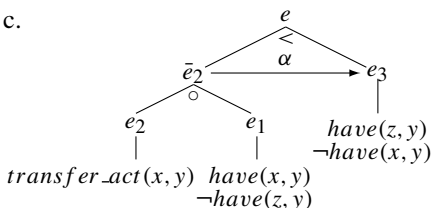
have(e_1, x, y)

¬have(e_1, z, y)

have(e_3, z, y)

¬have(e_3, x, y)

c.



Other modifications to VerbNet semantic representations also include the reinterpretation of accomplishments such as *build* and *write* in terms of GL's dynamic event semantics (Brown et al., 2019).

1.5 Conclusions

In this chapter, I have discussed the distinct but related paths taken toward the interpretation of events by the AI and linguistics communities. An early focus on planning and goal satisfaction in many of the major areas of interest in AI led to a concern with *plan compositionality*, for which the situation calculus became a major representational vehicle for reasoning and inference. Plans were viewed as traversals from *state* to *state*; hence, the primitive units in the language were situations describing state behavior and content. From the philosophically inspired linguistic community, Davidson's notion of event was motivated by a concern to model propositional content of a sentence, rather than an abstract plan. Hence, the focus over the past 50 years in linguistics has been on *semantic compositionality*. I describe how recent work in event semantics has attempted to bridge this cultural and representational divide. The approach taken in dynamic event semantics incorporates the language of state-change from AI with the compositional constraints on linguistic behavior from linguistics.

Acknowledgments This work was funded in part by DTRA grant DTRA-16-1-0002; DARPA grant FA8750-18-2-0016; and DARPA grant W911NF-15-C-0238. I thank Olga Batiukova, Martha Palmer, and Nikhil Krishnaswamy for commenting on an earlier version of this chapter. All remaining errors are, of course, my own.

References

- Allen, James. 1983. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, **26**(11), 832–843.
- Allen, James. 1984. Towards a General Theory of Action and Time. *Artificial Intelligence*, **23**, 123–154.
- Asher, Nicholas, and Lascardes, Alex. 2003. *Logics of Conversation*. Cambridge University Press.
- Bittar, André, Amsili, Pascal, Denis, Pascal, and Danlos, Laurence. 2011. French TimeBank: An ISO-TimeML Annotated Reference Corpus. Pages 130–134 of: *Proceedings of the 49th Annual Meeting of the Association for Computational*

- Linguistics: Human Language Technologies: Short Papers – Volume 2*. Association for Computational Linguistics.
- Bonial, Claire, Corvey, William, Palmer, Martha, Petukhova, Volha V., and Bunt, Harry. 2011. A Hierarchical Unification of LIRICS and VerbNet Semantic Roles. Pages 483–489 of: *2011 IEEE Fifth International Conference on Semantic Computing*. IEEE.
- Bowman, Samuel R., Angeli, Gabor, Potts, Christopher, and Manning, Christopher D. 2015. A Large Annotated Corpus for Learning Natural Language Inference. *arXiv preprint arXiv:1508.05326*.
- Brown, Susan Windisch, Bonn, Julia, Gung, James, et al. 2019. VerbNet Representations: Subevent Semantics for Transfer Verbs. Pages 154–163 of: *Proceedings of the First International Workshop on Designing Meaning Representations*.
- Brown, Susan Windisch, Pustejovsky, James, Zaenen, Annie, and Palmer, Martha. 2018. Integrating Generative Lexicon Event Structures into Verbnet. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Caselli, Tommaso, Lenzi, Valentina Bartalesi, Sprugnoli, Rachele, Pianta, Emanuele, and Prodanof, Irina. 2011. Annotating Events, Temporal Expressions and Relations in Italian: The It-TimeML Experience for the Ita-TimeBank. Pages 143–151 of: *Proceedings of the 5th Linguistic Annotation Workshop*. Association for Computational Linguistics.
- Chambers, Nathanael, and Jurafsky, Dan. 2008. Jointly Combining Implicit Constraints Improves Temporal Ordering. Pages 698–706 of: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Chambers, Nathanael, and Jurafsky, Dan. 2009. Unsupervised Learning of Narrative Schemas and Their Participants. Pages 602–610 of: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2*. Association for Computational Linguistics.
- Cheung, Jackie Chi Kit, Poon, Hoifung, and Vanderwende, Lucy. 2013. Probabilistic Frame Induction. *arXiv preprint arXiv:1302.4813*.
- Croft, William. 1991. *Syntactic Categories and Grammatical Relations: The Cognitive Organization of Information*. Chicago: University of Chicago Press.
- Dagan, Ido, Roth, Dan, Sammons, Mark, and Zanzotto, Fabio Massimo. 2013. Recognizing Textual Entailment: Models and Applications. *Synthesis Lectures on Human Language Technologies*, 6(4), 1–220.
- Davidson, Donald. 1967. The Logical Form of Action Sentences. *Essays on Actions and Events*, 5, 105–148.
- Dowty, David. 1979. *Word Meaning and Montague Grammar*. Dordrecht, The Netherlands: Kluwer.
- Emerson, E. Allen, and Sistla, A. Prasad. 1984. Deciding Full Branching Time logic. *Information and Control*, 61(3), 175–201.
- Fernando, Tim. 2009. Situations in LTL as Strings. *Information and Computation*, 207(10), 980–999.

- Fernando, Tim. 2013. Segmenting Temporal Intervals for Tense and Aspect. Pages 30–40 of: *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*.
- Finlayson, Mark A., Cremisini, Andreas, and Ocal, Mustafa. 2021. Extracting and Aligning Timelines. Pages 91–109 of: Caselli, Tommaso, Palmer, Martha, Hovy, Eduard, and Vossen, Piek (eds), *Computational Analysis of Storylines: Making Sense of Events*. Cambridge University Press.
- Goldman, Alvin I. 1970. *Theory of Human Action*. Princeton, NJ: Princeton University Press.
- Harel, David, Kozen, Dexter, and Tiuyun, Jerzy. 2000. *Dynamic Logic*. 1st ed. The MIT Press.
- Hobbs, Jerry R. 1982. Towards an Understanding of Coherence in Discourse. Pages 223–243 of: Hobbs, Jerry R., and Lehnert, W. (eds.), *Strategies for Natural Language Processing*. Lawrence Erlbaum.
- Hobbs, Jerry R. 1985. Ontological Promiscuity. Pages 60–69 of: *23rd Annual Meeting of the Association for Computational Linguistics*. Chicago: Association for Computational Linguistics.
- Hobbs, Jerry R., Croft, William, Davies, Todd, Edwards, Douglas, and Laws, Kenneth. 1987. Commonsense Metaphysics and Lexical Semantics. *Computational Linguistics*, **13**(3–4), 241–250.
- Hobbs, Jerry, and Pustejovsky, James. 2003. Annotating and Reasoning about Time and Events. In: *Proceedings of AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning (Stanford, CA, March 2003)*.
- Im, Seohyun, and Pustejovsky, James. 2010. Annotating Lexically Entailed Subevents for Textual Inference Tasks. Pages 204–209 of: Murray, R. Charles, and Guesgen, Hans W. (eds.), *Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference (FLAIRS-23)*. Daytona Beach, FL.
- Im, Seohyun, You, Hyunjo, Jang, Hayun, Nam, Seungho, and Shin, Hyopil. 2009. KTimeML: Specification of Temporal and Event Expressions in Korean Text. Pages 115–122 of: Riza, Hammam, and Sornlerlamvanich, Virach (eds.), *Proceedings of the 7th Workshop on Asian Language Resources*. Association for Computational Linguistics.
- Jackendoff, Ray. 1983. *Semantics and Cognition*. Vol. 8. The MIT Press.
- Jackendoff, Ray. 1990. *Semantic Structures*. Vol. 18. MIT Press.
- Kamp, J. A. W. 1968. *Tense Logic and the Theory of Linear Order*. Ph.D. Thesis, University of California, Los Angeles.
- Kipper, Karin, Korhonen, Anna, Ryant, Neville, and Palmer, Martha. 2008. A Large-Scale Classification of English Verbs. *Language Resources and Evaluation*, **42**(1), 21–40.
- Kowalski, Robert, and Sergot, Marek. 1989. A Logic-Based Calculus of Events. Pages 23–55 of: Schmidt, J. W., and Thanos, C. (eds.), *Foundations of Knowledge Base Management*. Berlin: Springer.
- Lakoff, George. 1970. *Irregularity in Syntax*. Holt, Rinehart, and Winston.
- Levin, Beth. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago press.
- Mani, Inderjeet, and Pustejovsky, James. 2012. *Interpreting Motion: Grounded Representations for Spatial Language*. Oxford University Press.

- McCarthy, John. 1963. *Situations, Actions, and Causal Laws*. Tech. Rep., Stanford University, Department of Computer Science.
- McCarthy, John, and Hayes, Patrick. 1969. *Some Philosophical Problems from the Standpoint of Artificial Intelligence*. Stanford University.
- McCawley, James D. 1971. Prelexical Syntax. *Monograph Series on Languages and Linguistics*, **24**, 19–33.
- McDermott, Drew. 1982. A Temporal Logic for Reasoning about Processes and Plans. *Cognitive Science*, **6**(2), 101–155.
- Moens, Marc, and Steedman, Mark. 1988. Temporal Ontology and Temporal Reference. *Computational Linguistics*, **14**, 15–28.
- Narayan-Chen, Anjali, Graber, Colin, Das, Mayukh, et al. 2017. Towards Problem Solving Agents That Communicate and Learn. Pages 95–103 of: Bansal, Mohit, Matuszek, Cynthia, Andreas, Jacob, Artzi, Yoav, and Bisk, Yonatan (eds.), *Proceedings of the First Workshop on Language Grounding for Robotics*.
- Naumann, Ralf. 2001. Aspects of Changes: A Dynamic Event Semantics. *Journal of Semantics*, **18**(1), 27–81.
- Parsons, Terence. 1989. The Progressive in English: Events, States and Processes. *Linguistics and Philosophy*, **12**(2), 213–241.
- Parsons, Terence. 1990. *Events in the Semantics of English*. Vol. 5. Cambridge, MA: MIT Press.
- Pichotta, Karl, and Mooney, Raymond. 2014. Statistical Script Learning with Multi-argument Events. Pages 220–229 of: Wintner, Shuly, Goldwater, Sharon, and Riezler, Stefan (eds.), *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*.
- Pnueli, Amir. 1977. The Temporal Logic of Programs. Pages 46–57 of: *18th Annual Symposium on Foundations of Computer Science (SFCS 1977)*. IEEE.
- Pratt-Hartmann, I. 2007. From TimeML to Interval Temporal Logic. Pages 166–180 of: Geertzen, J., Bunt, H. C., Thijsse, E., and Schiffrin (eds.), *Proceedings of the Seventh International Workshop on Computational Semantics (IWCS-7)*.
- Prior, A. N. 1957. *Time and Modality*. Oxford: Clarendon Press.
- Pustejovsky, James. 1988. The Geometry of Events. Tenny, C. (ed.), *Studies in Generative Approaches to Aspect*.
- Pustejovsky, James. 1991. The Syntax of Event Structure. *Cognition*, **41**(1), 47–81.
- Pustejovsky, James. 1995. *The Generative Lexicon*. Cambridge, MA: MIT Press.
- Pustejovsky, James. 2005 (August). *Generative Lexicon and Type Theory*. ESSLLI Summer School, Edinburgh, Scotland.
- Pustejovsky, James. 2013. Dynamic Event Structure and Habitat Theory. Pages 1–10 of: Pustejovsky, James (ed.), *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)*. Association for Computational Linguistics.
- Pustejovsky, James, Castao, José, Ingria, Robert, et al. 2003. TimeML: Robust Specification of Event and Temporal Expressions in Text. In: Bos, Johan, and Koller, Alexander (eds.), *Fifth International Workshop on Computational Semantics (IWCS-5)*.
- Pustejovsky, James, Lee, Kiyong, Bunt, Harry, and Romary, Laurent. 2010. ISO-TimeML: An International Standard for Semantic Annotation. In: Calzolari, Nicoletta, Choukri, Khalid, Maegaard, Bente (eds.), *Proceedings of the Seventh*

- International Conference on Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association.
- Pustejovsky, James, and Moszkowicz, Jessica L. 2011. The Qualitative Spatial Dynamics of Motion in Language. *Spatial Cognition & Computation*, **11**(1), 15–44.
- Reichenbach, Hans. 1947. *Elements of Symbolic Logic*. London: Collier-Macmillan.
- Tenny, Carol, and Pustejovsky, James. 2000. *Events as Grammatical Objects: The Converging Perspectives of Lexical Semantics and Syntax*. CSLI Publications, Center for the Study of Language and Information.
- UzZaman, Naushad, Llorens, Hector, Derczynski, Leon, et al. 2013. SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. Pages 1–9 of: Manandhar, Suresh, and Yuret, Deniz (eds.), *Second Joint Conference on Lexical and Computational Semantics (*SEM): Volume 2. Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta: Association for Computational Linguistics.
- van Benthem, Johannes Franciscus Abraham Karel. 1995. Logic and the Flow of Information. Pages 693–724 of: *Studies in Logic and the Foundations of Mathematics*, Vol. 134. Elsevier.
- Verhagen, Marc, Gaizauskas, Robert, Schilder, Frank, et al. 2007. Semeval-2007 Task 15: Tempeval Temporal Relation Identification. Pages 75–80 of: Agirre, Eneko, Màrquez, Lluís, and Wicentowski, Richard (eds.), *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*.
- Verhagen, Marc, Sauri, Roser, Caselli, Tommaso, and Pustejovsky, James. 2010. SemEval-2010 Task 13: TempEval-2. Pages 57–62 of: Erk, Katrin, and Strappara, Carlo (eds.), *Proceedings of the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.
- Vossen, Piek, Caselli, Tommaso, and Segers, Roxane. 2021. A Narratology-based Framework for Storyline Extraction. Pages 130–147 of: Caselli, Tommaso, Palmer, Martha, Hovy, Eduard, and Vossen, Piek (eds), *Computational Analysis of Storylines: Making Sense of Events*. Cambridge University Press.
- Zaenen, Annie, Bobrow, Daniel G., and Condoravdi, Cleo. 2008. The Encoding of Lexical Implications in VerbNet Predicates of Change of Locations. In: Calzolari, Nicoletta, Choukri, Khalid, Maegaard, Bente, et al. (eds.), *LREC*.