Triadic temporal representations and deformations

Tim Fernando
School of Computer Science & Statistics
Trinity College Dublin, Ireland
Tim.Fernando@tcd.ie

Abstract

Triadic representations that temporally order events and states are described, consisting of strings and sets of strings of bounded but refinable granularities. The strings are compressed according to J.A. Wheeler's dictum *it from bit*, with bits given by statives and non-statives alike. A choice of vocabulary and constraints expressed in that vocabulary shape representations of cause-and-effect with deformations characteristic, Mumford posits, of patterns at various levels of cognitive processing. These deformations point to an ongoing process of learning, formulated as grammatical inference of finite automata, structured around Goguen and Burstall's institutions.

1 Introduction

What does a string $\mathfrak{s}$ that is assigned a probability by a language model describe? Over a range of uses, $\mathfrak{s}$ is uttered at time S to describe an event occurrence at time E. Reichenbach (1947) suggests that S is connected to E by a *reference time* R, traversing three corners

language (S), agent (R), and world (E)

of a triangle that is arguably congruent with the well-known symbol-thought-referent triangle from Ogden and Richards (1923), page 11. Reichenbach derives nine *fundamental forms*, including the *simple past* (1) and *present perfect* (2), by positioning R relative to S and to E (with $<$ as "earlier than").

$$R < S \text{ and } R = E \qquad \text{(Ed ate.)} \qquad (1)$$

$$S = R \text{ and } E < R \qquad \text{(Ed has eaten.)} \qquad (2)$$

For fundamental forms, S, R and E may be considered points; but for extended tenses with the present participle (*-ing*) and temporal adverbs (such as *yesterday*), E and R are stretched to temporal intervals. E and R have since been refined in various ways (e.g., Moens and Steedman, 1988; Kamp and Reyle, 1993; Nelken and Francez, 1995; Asher and Lascarides, 2003; Klein, 2009; Kehler, 2022), and the speech time S extended to an interval timing an utterance event so that

	language	agent	world
Ogden and Richards, 1923	symbol	thought	referent
Reichenbach, 1947	S	R	E
Liang and Potts, 2015	u	s	d
Goguen and Burstall, 1992	*Sen*	**Sig**	**Mod**

Table 1: Some triads

(†) the meaning $[\![\mathfrak{s}]\!]$ of a simple declarative sentence $\mathfrak{s}$ is a relation $u\ [\![\mathfrak{s}]\!] e$ between an utterance u (with time S) and a described situation e (with time E).

(†) is an early formulation of a *relation theory of meaning* (Barwise and Perry, 1983, page 19) that is developed further in, for example, Cooper and Ginzburg (2015); Cooper (2023). Left out of (†) is the reference time R which Reichenbach uses as a bridge between S and E. That is, (†) is dyadic, supplying an utterance u and denotation $d = e$ for a *linguistic object* $\langle u, s, d\rangle$ in Liang and Potts (2015) with*out* a semantic representation s which Table 1 aligns with R (under *agent* and *thought*, sandwiched between the utterance u of the string $\mathfrak{s}$, and the denotation[1] d).

Among the semantic representations considered in Liang and Potts (2015) are "distributed representations — vectors and matrices" that feed into string probabilities from language models. The present work focuses on semantic representations that support probabilities via more familiar logical forms. These forms describe patterns of events that are linked below to *pattern theory* (Grenander and Miller, 2007), which D. Mumford defines as

> the analysis of the patterns generated by the world in any modality, with all their

[1] (†)'s denotation e is closer to the "conventional meaning" than to the "communicative intent" discussed in Bender and Koller (2020)'s critique of large neural language models.

Proceedings of the 4th Natural Logic Meets Machine Learning Workshop (NALOMA23), pages 51–61
June 20, 2023.

> naturally occurring complexity and ambiguity, with the goal of reconstructing the processes, objects and events that produced them. [Mumford, 1994, page 187]

Expecting such a link flies against Mumford's view that pattern theory "stands in opposition to the accepted analysis of thought in terms of logic" but is less surprising if indeed "pattern theory contains the germs of a universal theory of thought itself" [page 221]. Fundamental to pattern theory is a "principle of realism" stating that

> the pattern should not merely describe the 'pure' situation that underlies reality but the 'deformed' situation that is actually observed in which the pure pattern may be hard to recognize. This generalizes, for example, Chomsky's idea of the deep structure of an utterance vs. its surface structure, where deep $\sim$ pure and surface $\sim$ deformed. [Mumford, 2019, page 203]

Mumford (1994) claims "the world does not have an infinite repertoire of different tricks which it uses to disguise what is going on" and picks out four types of deformations "encountered at all levels of cognitive processing." These deformations can be seen in cause-and-effect representations formed below which share two basic features with the information-theoretic formulation of pattern theory in Mumford (1994)

(i) a finite space Ω of functions f from a finite set of variables to a finite set of values, and

(ii) an encoding of f such that code(f) has a length which is minimized to reconstruct the world w that f is about.

By restricting to finite sets, (i) bounds the granularity of the representation, imposing a finite precision on values. The blurring here is an instance of one of Mumford's four types of deformations, taken up in section 3 below, where it is associated with a move from records to record types (Cooper and Ginzburg, 2015; Cooper, 2023). The code lengths mentioned in (ii) are used in Mumford (1994) for an approach to Bayesian maximum likelihood estimates based on Shannon's optimal coding theorem. The function f and the world w it is about in (ii) can, from the perspective of Table 1 above, be likened to an utterance u and denotation d that u is about. Even for simple declarative sentences $\mathfrak{s}$, however, the leap from an utterance u of $\mathfrak{s}$ to its denotation d is an enormous one, inviting the question: would a mediating representation s between u and d not help? Arguably, such a representation s is what code(f) in (ii) is, although it is not obvious from Mumford (1994) or Grenander and Miller (2007) what form s might take for an utterance u of a declarative sentence.

The semantic representations s below describe not only events such as denotations d but also utterances u of pieces $\mathfrak{s}$ of language ranging from multi-sentential discourses (as in Kamp and Reyle (1993)) down to subsentential units. Following (ii), code lengths are minimized in section 2, but appealing in this case to Wheeler (1990)'s dictum *it from bit*. To illustrate the idea, consider Reichenbach's simple past (1) and present perfect (2), reformulated as strings $\boxed{E,R}\boxed{S}$ and $\boxed{E}\boxed{R,S}$ respectively, both of length 2, and the past perfect (3) represented by the string $\boxed{E}\boxed{R}\boxed{S}$ of length 3.[2]

$$R < S \text{ and } E < R \qquad \text{(Ed had eaten.)} \qquad (3)$$

If we focus on S and E and throw R out, we can compress all three strings to $\boxed{E}\boxed{S}$ representing the relation $E < S$ common to (1), (2) and (3), saying u is about an event d in u's past. The details are given in section 2, where strings are formulated as models of predicate logic (e.g., Libkin, 2004) with a specified signature fixing granularity. While that granularity is bounded by finite sets in (i), ever larger finite signatures Σ can be collected in a category **Sig** that a functor **Mod** maps contravariantly to sets of Σ-models and a functor *Sen* maps covariantly to sets of Σ-sentences. The triad *Sen*, **Sig**, **Mod** occupies the bottom row of Table 1, and can be organised into a logical system called an *institution* (Goguen and Burstall, 1992; Goguen, 2006). An amalgamation property enjoyed by well-behaved *algebraic* institutions (e.g., Sannella and Tarlecki, 2015) is, however, damaged by compression. This is explained in section 3, where compression is equated with another of Mumford (1994)'s deformations, domain warping. Further deformations are noted that shape the sample space Ω on which a probability measure is defined (yielding probabilities that are front and center in pattern theory). What makes the strings here interesting is

[2] $\boxed{E,R}\boxed{S}$ is written E, R-S in Reichenbach (1947). Using boxes instead of curly braces $\{,\}$ for sets qua string symbols suggests reading a comic strip (e.g., Fernando, 2015).

	active	stative
Kleene, 1956	input cell	inner cell
dynamic logic	program	proposition
action language	elementary action	fluent
sig (A, V)	act $\in A$	variable $\in dom(V)$

Table 2: Deconstructing a transition $q \xrightarrow{a} q'$

that they represent some of "processes, objects and events" that produce patterns. These patterns include certain causes and effects, packaged as *event nuclei* in Moens and Steedman (1988), that can be framed around transitions in finite automata amenable to probabilistic elaboration.

2 Strings as compressed models

The neural nets for which Kleene (1956) introduced finite automata have cells of two kinds: *input cells* which could either fire or not, and *inner cells* which could take one of finitely many values, depending on the input cells and inner cells that feed into them. For neural nets with k input cells $\mathcal{N}_1, \ldots, \mathcal{N}_k$, Kleene forms an alphabet of 2^k symbols (one for each subset of $\{\mathcal{N}_1, \ldots, \mathcal{N}_k\}$), and from m inner cells $\mathcal{M}_1, \ldots, \mathcal{M}_m$, generates m-tuples $(v_1, \ldots, v_m)$ consisting of values v_i that $\mathcal{M}_i$ can take. A couple of notational conventions will prove handy below. For any integer $j > 0$, let us write $[j]$ for the set of j integers from 1 to j

$$[j] \ := \ \{1, 2, \ldots, j\}.$$

Next, given a set-valued function V, let $\prod V$ be the set of V-records, where a *V-record* is a function r with the same domain as V that maps each $\mathbf{x}$ in $dom(V)$ to an element $r(\mathbf{x})$ of $V(\mathbf{x})$. It is often convenient to write $\prod V$ out as $\prod_{\mathbf{x} \in dom(V)} V(\mathbf{x})$. For example, if each inner cell $\mathcal{M}_i$ can take s_i many values, then the set

$$\prod_{i \in [m]} [s_i] \ \cong \ [s_1] \times \cdots \times [s_m]$$

of functions r mapping $i \in [m]$ to one of s_i many values, $r(i)$, is isomorphic to the set of m-tuples $(v_1, \ldots, v_m)$ assigning inner cell $\mathcal{M}_i$ the v_i^{th} of s_i values. $\prod_{i \in [m]} [s_i]$ can serve as the set of states between which any set $a \subseteq \{\mathcal{N}_1, \ldots, \mathcal{N}_k\}$ of input cells can label a binary relation $\xrightarrow{a}$ of transitions $q \xrightarrow{a} q'$ from state q to q'. Table 2 aligns inner cells with the stative sides q, q' of $q \xrightarrow{a} q'$, and input cells with the active middle a. The stative/active dichotomy is perhaps most famously developed in the proposition/program distinction drawn in *Dynamic Logic* (Harel et al., 2000), but the conception of a transition label a as a set of firing input cells puts us on a different course.

Input cells become elementary actions in *action languages* (Gelfond and Lifschitz, 1998), where a transition label a (called an action) is a set of elementary actions, while a state q is described by values taken by certain fluents[3] corresponding to inner cells. See the penultimate row of Table 2. The bottom row Table 2 brings out what Kleene (1956) and Gelfond and Lifschitz (1998) have in common through the following rudimentary notion of *sig*nature.

Definition. A *sig* is a pair (A, V) consisting of a finite set A of *acts* and a function V with a finite domain, $dom(V)$, of *variables* $\mathbf{x}$, each paired with a finite set $V(\mathbf{x})$ of *values* that $\mathbf{x}$ can take.

A sig (A, V) provides a finite vocabulary of acts in A and statives (given by variables and values) in V. Statives are central to works such as Dowty (1979), where they are the basis of an aspectual calculus.

An instructive example is provided by the leap below from (4) to (5) by virtue of the entailment (6) from *bought* to *owns* proposed in Hosseini (2020); see also Hosseini et al. (2019).

Facebook bought Instagram (4)

Facebook owns Instagram (5)

$$\boxed{\text{bought}(x, y)} \Rightarrow \boxed{\text{owns}(x, y)} \tag{6}$$

(6) assumes no change in ownership of y after x bought y; this assumption may fail depending on subsequent events. (7) repairs this flaw in (6) by applying the operator BECOME to the (untensed) stative own(x, y) to produce a non-stative BECOME(own(x, y)).

$$\boxed{\text{buy}(x, y)} \Rightarrow \boxed{\text{BECOME}(\text{own}(x, y))} \tag{7}$$

The meaning of BECOME in (7) is brought out in a transition (8) labelled by buy(x, y) from a state where x does *not* own y to a state where x does (0 marking falsity, and 1 truth).

$$\boxed{(\text{own}(x, y), 0)} \xrightarrow{\text{buy}(x,y)} \boxed{(\text{own}(x, y), 1)} \tag{8}$$

[3] Action languages belong to the symbolic AI tradition blazed by John McCarthy, who adopted Newton's term *fluent* for a state variable (the value of which may change over time).

Entailments such as (9), however, make clear there is more to buy(x,y) than BECOME(own(x,y)).

$$\boxed{\text{buy}(x,y)} \Rightarrow \boxed{\text{pay}(x,y)} \tag{9}$$

It is easy enough to replace buy(x,y) in (8) by BECOME(own(x,y)), but the question is: can we reduce (4) to a transition $q \xrightarrow{a} q'$ without leaving out some of the details, such as *Facebook paid for Instagram*, implicit in (4) according to (9)? There are two directions along which to extend $q \xrightarrow{a} q'$. First, more than one act may go into the transition label a on the understanding that

(‡) $q \xrightarrow{a} q'$ says: the acts in a execute concurrently to move from q to q'.

Second, we may break $q \xrightarrow{a} q'$ down to a chain (10) of n transitions $q_{i-1} \xrightarrow{a_i} q_i$ between states q_{i-1} and q_i labelled by sets a_i of acts from $q_0 = q$ to $q_n = q'$.

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots \xrightarrow{a_n} q_n. \tag{10}$$

Now, for any fixed n in (10), is it not conceivable that some (if not every) transition $q_{i-1} \xrightarrow{a_i} q_i$ can be refined to a longer transition chain from q_{i-1} to q_i? Perhaps so. But if we use a sig (A,V) to require of an (A,V)-chain (10) that

$$q_0, q_i \text{ are } V\text{-records} \quad \text{and} \quad a_i \subseteq A \tag{11}$$

then it is more plausible that further refinements of (10) would involve stepping from the sig (A,V) to a suitably larger sig (A',V'). Just what suitably larger means, we take up in the next section. In the meantime, note that the transition (8) serves as an account of buy(x,y) for the sig (A,V), where A is $\{\text{buy}(x,y)\}$ and the function V is, as a set of ordered pairs $(\mathbf{x}, V(\mathbf{x}))$, the singleton

$$V = \{(\text{own}(x,y), \{0,1\})\}$$

with exactly one variable own(x,y), the values of which are either 0 or 1.

Next, fixing a sig (A,V), let us package the (A,V)-chain (10) as the $(A.V)$-string

$$(q_0,a_1)(q_1,a_2)\cdots(q_{n-1},a_n)(q_n,a_{n+1})$$

of $n+1$ pairs (q_{i-1},a_i), where $a_{n+1} = \emptyset$. To simplify notation, let us assume

(NAP) **n**o **a**ct in A is an ordered **p**air

so that given a pair (q,a) of $q \in \prod V$ and $a \subseteq A$, we can recover from the union $\alpha = q \cup a$, the label a and state q

$$a = \alpha \cap A \quad \text{and} \quad q = \alpha \setminus A$$

through set complementation

$$X \setminus Y \;:=\; \{x \in X \mid x \notin Y\}.$$

Flattening (q,a) to $q \cup a$, the transition (8) becomes the string

$$\boxed{(\text{own}(x,y),0),\, \text{buy}(x,y)}\boxed{(\text{own}(x,y),1)}$$

of length 2, the first symbol/box of which has stative part $\boxed{(\text{own}(x,y),0)}$ and active part $\boxed{\text{buy}(x,y)}$. The partiality of a sig suggests widening the range of (A,V)-strings beyond those obtained from transition chains (10) that (11) ties to a sig (A,V). We drop (11) to accommodate larger sigs (A',V') where q_0, q_i are V'-records and $a \subseteq A'$. In particular, we might extract the $(A,\emptyset)$-string $a_1 a_2 \cdots a_n$ from (10) where each of the labels a_i is a subset of A. Adding the restriction that each a_i be non-empty leads us to Durand and Schwer (2008), where an *S-word* is defined to be a string of non-empty sets. But why exclude the empty box $\square$ from an S-word?

Under (‡), it is natural to assume the value of a variable cannot change without an act, leading to the following principle of inertia

$$\text{whenever } q \xrightarrow{\square} q',\; q = q'. \tag{12}$$

The transition $q \xrightarrow{\square} q$ hardly describes any change, and arguably carries zero information, suggesting that any occurrence of the empty box $\square$ in $a_1 \cdots a_n$ be deleted. Similarly, if we extract the $(\emptyset,V)$-string $q_0 q_1 \cdots q_n$ of states from (10) for a sig (A,V) where each q_i in (10) is a V-record. then any stutter qq might be deleted from $q_0 q_1 \cdots q_n$, as in the *block compression* $\mathit{bc}(s)$ *of* a string s

$$\mathit{bc}(s) \;:=\; \begin{cases} s & \text{if length}(s) \leq 1 \\ \mathit{bc}(qs') & \text{if } s = qqs' \\ q\,\mathit{bc}(q's') & \text{if } s = qq's' \\ & \quad \text{where } q \neq q' \end{cases}$$

(Fernando, 2015).

For sigs (A,V) where neither A nor V need be empty, let us collect the (A,V)*-boxes* from which we form (A,V)-strings in the alphabet

$$\mathcal{B}_{A,V} \;:=\; \{a \cup r \mid a \subseteq A \text{ and } r \in \prod V\}$$

and define the *A-compression $\kappa_A(s)$ of* a string $s \in \mathcal{B}_{A,V}{}^*$ by induction on the length of s

$$\kappa_A(s) \ := \begin{cases} \epsilon & \text{if } s = \epsilon \text{ or } s = \Box \\ s & \text{else if length}(s) = 1 \end{cases}$$

and for strings of length ≥ 2,

$$\kappa_A(\alpha\alpha' s) \ := \begin{cases} \kappa_A(\alpha' s) & \text{if } \alpha = \Box \text{ or} \\ & \ \alpha = \alpha' \setminus A \\ \alpha \, \kappa_A(\alpha' s) & \text{otherwise} \end{cases}$$

(Fernando, 2022). Clearly,

$$\kappa_A(\kappa_A(s)) = \kappa_A(s)$$

and in case A or V is empty,

$$\kappa_A(s) \ = \begin{cases} s \text{ without } \Box\text{'s} & \text{if } V = \emptyset \\ bc(s) & \text{else if } A = \emptyset. \end{cases}$$

A-compression κ_A implements the Aristotelian dictum *no time without change* (e.g., Coope, 2001), or better:

$$\text{no time}_{A,V} \text{ without change}_{A,V}. \tag{13}$$

To see this, it is useful to construe a non-empty string of (A, V)-boxes as a *model of Monadic Second-Order* logic (MSO), with MSO-sentences that capture the sets of such strings accepted by finite automata via a satisfaction relation $\models$ (e.g., Libkin, 2004, Theorem 7.21).

More precisely, let us collect the possible input/output pairs of V-records in the set

$$\sum V \ := \ \{(\mathbf{x}, c) \mid \mathbf{x} \in dom(V) \text{ and } c \in V(\mathbf{x})\}.$$

Let the *vocabulary* of a sig (A, V) be the union

$$voc(A, V) \ := \ A \cup \sum V$$

of A with $\sum V$, and for every $u \in voc(A, V)$, let us form a fresh unary relation symbol P_u. P_u is interpreted relative to a string $\alpha_1 \cdots \alpha_n \in \mathcal{B}^+_{A,V}$ of (A, V)-boxes α_i as the set

$$[\![P_u]\!]_{\alpha_1 \cdots \alpha_n} \ := \ \{i \in [n] \mid u \in \alpha_i\}$$

of string positions i where u occurs. Hence, the disjunction $\bigvee_{u \in A} P_u(i)$ says: some act from A occurs at i. In addition to unary relation symbols P_u, there is a binary relation S that is interpreted as the successor (+1) relation

$$[\![S]\!]_{\alpha_1 \cdots \alpha_n} \ := \ \{(i, i+1) \mid i \in [n-1]\}$$

on string positions. We can conjoin $P_u(i)$ with the negation of the claim that u occurs at a successor of i for the formula

$$\delta_u(i) \ := \ P_u(i) \wedge \neg\exists j(iSj \wedge P_u(j))$$

which for $u \in \sum V$ can be paraphrased: u holds at position i but not immediately afterwards. Accordingly, the MSO-sentence

$$ntwoc_{A,V} \ := \forall i \ (\bigvee_{u \in A} P_u(i) \vee \bigvee_{u \in \sum V} \delta_u(i))$$

says:

> at every string position, some act from A occurs or some V-stative holds but not immediately afterwards

which amounts to (13), assuming string positions represent time$_{A,V}$, and change$_{A,V}$ is communicated through the set

$$\{P_u \mid u \in A\} \ \cup \ \{\delta_u \mid u \in \sum V\} \tag{14}$$

of active predicates P_u ($u \in A$) and stative changes δ_u ($u \in \sum V$). It turns out $ntwoc_{A,V}$ expresses the effect of A-compressing strings over the alphabet $\mathcal{B}_{A,V}$.

Theorem. *For all $s \in \mathcal{B}_{A,V}{}^+$,*

$$s \models ntwoc_{A,V} \iff s = \kappa_A(s).$$

The theorem is proved by a routine induction on the length of s. Our account of patterns based on (A, V) will center around the set

$$\mathbf{Mod}(A, V) \ := \{\kappa_A(s) \mid s \in \mathcal{B}_{A,V}{}^+\}$$

of *(A, V)-models*, which the theorem above equates with (A, V)-strings satisfying $ntwoc_{A,V}$

$$\mathbf{Mod}(A, V) \ = \{s \in \mathcal{B}_{A,V}{}^+ \mid s \models ntwoc_{A,V}\}.$$

There are two kinds of variables here against which to apply Quine (1950)'s prescription that

> to be assumed as an entity is to be assumed as a value of a variable (p. 228)

— viz., so-called variables $\mathbf{x}$ in $dom(V)$ with values in $V(\mathbf{x})$, and variables such as i, j that occur free and bound in $\delta_u(i)$, and range over time. The latter time variables link $ntwoc_{A,V}$ to J.A. Wheeler's dictum *it from bit* every it — every particle, every field of force, even the space**time** continuum itself — derives its function, its meaning, its very existence entirely — even if in some contexts indirectly — from the apparatus-elicited answers to yes-or-no questions, binary choices, bits. [Wheeler, 1990, p.5]

The string positions $[n]$ of $\alpha_1 \cdots \alpha_n$ are constrained by $ntwoc_{A,V}$ to changes$_{A,V}$ (14) observed through the apparatus MSO$_{A,V}$. The power of that apparatus is bound by the sig (A, V) which is refined in the next section to expand what can be observed, uncovering deformations along the way.

3 Projections and deformations

Relaxing the finiteness assumptions built into a sig, let us fix a pair $(\mathsf{Act}, \mathsf{Val})$ of

(a) a set Act of acts, none of which is an ordered pair (building in the no-act-pair assumption (NAP) from the previous section), and

(b) a function Val from variables $\mathbf{x}$ to sets $\mathsf{Val}(\mathbf{x})$ of values that $\mathbf{x}$ can take.

A *finite blurring of* Val is a function V whose domain, $dom(V)$, is a finite subset of $dom(\mathsf{Val})$ such that for each $\mathbf{x} \in dom(V)$, $V(\mathbf{x})$ is a finite partition of $\mathsf{Val}(\mathbf{x})$. Thus, $\sum V$ is finite even if $\sum \mathsf{Val}$ is not (due to $dom(\mathsf{Val})$ or some $\mathbf{x} \in dom(\mathsf{Val})$ with infinite $\mathsf{Val}(\mathbf{x})$). The intuition is that V approximates Val up to finite precision.[4] Under *it-from-bit*, the finite approximations V have an arguably stronger claim to reality than the idealization Val.

With this in mind, let us define an $(\mathsf{Act}, \mathsf{Val})$-*sig* to be a pair (A, V) of a finite subset A of Act and a finite blurring V of Val. $(\mathsf{Act}, \mathsf{Val})$-sigs can be partially ordered as follows. (A, V) *is refined by* (A', V'), written $(A, V) \preceq (A', V')$, if $A \subseteq A'$, $dom(V) \subseteq dom(V')$ and for each $\mathbf{x} \in dom(V)$, the partition $V'(\mathbf{x})$ refines $V(\mathbf{x})$ in the usual sense (i.e., every value-set from $V'(\mathbf{x})$ is a subset of some value-set from $V(\mathbf{x})$). Assuming $(A, V) \preceq (A', V')$, let

(a) the (A, V)-*reduct of* an (A', V')-box α' be the (A, V)-box

$$\rho_{A,V}(\alpha') \;:=\; (\alpha' \cap A) \cup \alpha'_V$$

where $\alpha'_V \in \prod V$ maps $\mathbf{x} \in dom(V)$ to the unique $V(\mathbf{x})$-equivalence class that includes the value-set that α' assigns to $\mathbf{x}$[5]

(b) the (A, V)-*reduct of* a string of (A', V')-boxes be its componentwise (A, V)-reduct

$$\rho_{A,V}(\alpha'_1 \cdots \alpha'_n) \;:=\; \rho_{A,V}(\alpha'_1) \cdots \rho_{A,V}(\alpha'_n)$$

(c) the (A, V)-*projection of* an (A', V')-model s' be the A-compression of its (A, V)-reduct

$$\kappa_{A,V}(s') \;:=\; \kappa_A(\rho_{A,V}(s'))\ .$$

For example, given an (A', V')-model $\alpha'_1 \cdots \alpha'_n$, its $(\emptyset, V)$-projection for $V \neq \emptyset$ is the block compression

$$bc((\alpha'_1)_V \cdots (\alpha'_n)_V)$$

and its $(A, \emptyset)$-projection is the S-word

$$(\alpha'_1 \cap A) \cdots (\alpha'_n \cap A) \text{ without } \square\text{'s.}$$

Returning to Reichenbach's fundamental forms, if we treat the points E, R, S as acts, then $\boxed{E}\boxed{S}$ is the $(\{E, S\}, \emptyset)$-projection of each of the strings

$$\boxed{E,R}\boxed{S},\ \boxed{E}\boxed{R,S},\ \boxed{E}\boxed{R}\boxed{S}$$

for the simple past (1), present perfect (2) and past perfect (3), respectively. Shortening $\boxed{E}\boxed{R}\boxed{S}$ to $\boxed{E}\boxed{S}$ is an instance of *domain warping* (Mumford, 1994, p. 196) inasmuch as the domain of a string, as an MSO-model, is its set of string positions. In general, any change in string length from s' to $\kappa_{A,V}(s')$ can be put down to the compression κ_A built into $\kappa_{A,V}$. A-compression is required by the Theorem from the previous section if an (A, V)-model is to satisfy $ntwoc_{A,V}$. It is also indispensible for representing finite subsets of the real line $\mathbb{R}$ (a popular model of time) as strings of finite length — e.g., $\{0, 1, e, \pi\} \subseteq \mathbb{R}$ as $\boxed{0}\boxed{1}\boxed{e}\boxed{\pi}$ depicting $0 < 1 < e < \pi$. Clearly, $\mathbb{R}$ can be reconstructed by a projective (inverse) limit over string representations of its finite subsets. Take away A-compression and we lose this reconstruction.

Unfortunately, A-compression complicates the amalgamation of different (A, V)-projections.

[4]The reduction of $dom(\mathsf{Val})$ to a finite subset $dom(V)$ is compatible with the usual restriction on records to finitely many fields; the blurring of values in $\mathsf{Val}(\mathbf{x})$ to subsets of $\mathsf{Val}(x)$ in $V(\mathbf{x})$ suggests a further move to record types (Cooper and Ginzburg, 2015; Cooper, 2023).

[5]When V is V' restricted to $dom(V)$ (i.e., $V \subseteq V'$), the (A, V)-reduct $\rho_{A,V}(\alpha')$ of α' is just the intersection $\alpha' \cap voc(A, V)$.

This can be seen by looking once more at Reichenbach (1947)'s fundamental forms. Inasmuch as $R < S$ can be pictured as $\boxed{R \mid S}$ and $R = E$ as $\boxed{E, R}$, the step from the conjunction (1) of $R < S$ and $R = E$ to the string $\boxed{E, R \mid S}$ can be expressed as

$$\boxed{R \mid S} \,\&\, \boxed{E, R} \;=\; \boxed{E, R \mid S}.$$

On the other hand, the conjunction of $R < S$ and $R < E$ for the *posterior past* yields three different strings

$$\boxed{R \mid E, S}\,,\quad \boxed{R \mid E \mid S}\,,\quad \boxed{R \mid S \mid E}\,, \tag{15}$$

each of which has $(\{R, S\}, \emptyset)$-projection $\boxed{R \mid S}$ for $R < S$ and $(\{R, E\}, \emptyset)$-projection $\boxed{R \mid E}$ for $R < E$. (Similarly, for the *anterior future* from $S < R$ ad $E < R$). The non-uniqueness here can be summarized as

($*$) the presheaf **Mod** does not satisfy the gluing condition necessary for a sheaf

which we presently unpack. **Mod** is a *presheaf* insofar as **Mod** can be understood as a set-valued contravariant functor from the category **Sig** of (A, V)-sigs with morphisms given by the ordered pairs $((A, V), (A', V'))$ from refinement $\preceq$, where $\mathbf{Mod}((A', V'), (A, V))$ maps an (A', V')-model s' to its (A, V)-projection $\kappa_{A,V}(s')$

$$\text{i.e., } \mathbf{Mod}((A', V'), (A, V))(s') = \kappa_{A,V}(s').$$

Next, let us call two $(\mathsf{Act}, \mathsf{Val})$-sigs (A_1, V_1) and (A_2, V_2) *compatible* if V_1 and V_2 agree on the intersection of their domains

$$\text{i.e., } (\forall \mathbf{x} \in dom(V_1) \cap dom(V_2))\ V_1(\mathbf{x}) = V_2(\mathbf{x})$$

making $(A_1 \cup A_2, V_1 \cup V_2)$ an $(\mathsf{Act}, \mathsf{Val})$-sig. Given compatible sigs (A_1, V_1) ad (A_2, V_2), and (A_i, V_i)-models s_i for $i \in [2]$, let $s_1 \& s_2$ be the set of all $(A_1 \cup A_2, V_1 \cup V_2)$-models s that project to s_1 and to s_2

$$\kappa_{A_1,V_1}(s) = s_1 \text{ and } \kappa_{A_2,V_2}(s) = s_2.$$

The gluing condition in ($*$) requires that the set $s_1 \& s_2$ be a singleton whenever s_1 and s_2 agree on the $(\mathsf{Act}, \mathsf{Val})$-sig $(A_1 \cap A_2, V_1 \cap V_2)$

$$\kappa_{A_1 \cap A_2, V_1 \cap V_2}(s_1) = \kappa_{A_1 \cap A_2, V_1 \cap V_2}(s_2).$$

This requirement is not met by $\boxed{R \mid S} \& \boxed{R \mid E}$, which consists of the three strings in (15).[6] Only the first string $\boxed{R \mid E, S}$ would remain were we to drop A-compression from (A, V)-projection.[7]

Keeping A-compression, we shall give an account of the conjunction $s_! \& s_2$ above through a functor *Sen* from the category **Sig** mapping an $(\mathsf{Act}, \mathsf{Val})$-sig (A, V) covariantly to a set $Sen(A, V)$ of (A, V)-*sentences*. There are as many choices of $Sen(A, V)$ as there are ways of defining the languages accepted by finite automata, the crucial requirement on $Sen(A, V)$ being that there be a relation $\models_{A,V}$ between (A, V)-models and (A, V)-sentences such that

(i) for every (A, V)-sentence φ, there is a finite automaton accepting the set

$$\mathbf{Mod}_{A,V}(\varphi) := \{s \in \mathbf{Mod}(A, V) \mid s \models_{A,V} \varphi\}$$

of (A, V)-models that satisfy φ (under $\models_{A.V}$)

and conversely,

(ii) for every subset L of $\mathbf{Mod}(A, V)$ that is accepted by some finite automaton, there is some (A, V)-sentence φ capturing L

$$L = \mathbf{Mod}_{A,V}(\varphi).$$

For concreteness, we may equate $Sen(A, V)$ with the set of $\mathrm{MSO}_{voc(A,V)}$-sentences. Now, whenever $(A, V) \preceq (A', V')$, let $Sen((A, V), (A', V'))$ map an (A, V)-sentence φ to an (A', V')-sentence $\langle (A, V), (A', V') \rangle \varphi$ such that

($**$) $\mathbf{Mod}_{A',V'}(\langle (A, V), (A', V') \rangle \varphi)$ is the set

$$\{s' \in \mathbf{Mod}(A', V') \mid \kappa_{A,V}(s') \models_{A,V} \varphi\}$$

of (A', V')-models whose (A, V)-projections satisfy φ.

($**$) is the *Satisfaction condition* characteristic of an *institution* (Goguen and Burstall, 1992). The existence of an (A', V')-sentence $\langle (A, V), (A', V') \rangle \varphi$ validating ($**$) follows from the regularity assumptions (i) and (ii) above, and

[6] In this case, $A_1 = \{R, S\}, A_2 = \{R, E\}, V_1 = V_2 = \emptyset$. In general, $(A, V) \preceq (A', V')$ implies $A \subseteq A'$ but not necessarily $V \subseteq V'$. To sidetep notational complications, however, our discussion of gluing will proceed with the simple case of $V \subseteq V'$.

[7] Gluing is referred to as *amalgamation* in, for example, Sannella and Tarlecki (2015), where it is admitted by algebraic institutions with reducts as projections.

the closure of regular languages under inverse images of relations such as $\kappa_{A,V}$ computed by finite-state transducers. Under $(**)$, $\langle(A,V),(A',V')\rangle$ is a modal operator for $\kappa_{A,V}$, albeit not one of the primitive propositional connectives or quantifiers in MSO. Now, given two compatible sigs (A_1,V_1) and (A_2,V_2) and two (A_i,V_i)-sentences φ_i for $i \in [2]$, let us attach the modal operator $\langle(A_i,V_i),(A_1\cup A_2,V_1\cup V_2)\rangle$ to φ_i for

$$\psi_i \ := \ \langle(A_i,V_i),(A_1\cup A_2,V_1\cup V_2)\rangle\varphi_i$$

and observe that the conjunction $\psi_1\wedge\psi_2$ captures $s_1\&s_2$ provided φ_i captures s_i for $i\in[2]$. Such a conjunction is an instance of *multi-scale superposition* (Mumford, 1994, p. 195), the third of four types of deformations instantiated above (alongside blur, $\preceq$, and domain warping, $\kappa_{A,V}$).

The fourth of Mumford's deformations arises when examining cause-and-effect within a sig (A,V). For a handle on how acts $u\in A$ affect statives $v\in\sum V$, let us fix a function af with domain Act mapping every act $u\in\mathsf{Act}$ to a set $\mathsf{af}(u)\subseteq dom(\mathsf{Val})$ of variables that u can *af*fect. Given a pair $(\mathbf{x},c)\in\sum V$, let us collect the acts in A that can affect $\mathbf{x}$ in

$$A_{(\mathbf{x},c)} \ := \ \{u\in A \mid \mathbf{x}\in\mathsf{af}(u)\}.$$

Next, we form an MSO-formula $\delta_v(i,j)$ saying v holds at i but not at its successor j

$$\delta_v(i,j) \ := \ iSj\wedge P_v(i)\wedge\neg P_v(j).$$

Building on our understanding $(\ddagger)$ of transitions $q\xrightarrow{a}q'$ and inertia (12) from section 2, let us agree that an (A,V)-model s is af-*inertial* if for every pair $v\in\sum V$,

$$\forall i\forall j\ \ (\delta_v(i,j)\supset\bigvee_{u\in A_v}P_u(i)) \qquad (16)$$

which is to say: any v-change in s occurs with an act in A that can affect v. One of the challenges in meeting (16) is that the act that affects v need not be in the finite subset A of Act. Indeed, an af-inertial string s may, for some $A_\circ\subset A$, have $(A_\circ,V)$-projection $\kappa_{A_\circ,V_\circ}(s)$ that is *not* af-inertial because (16) requires an act $u\in A\setminus A_\circ$ outside $A_\circ$. (A,V)-models s which are not af-inertial are "incompelete observations" called "interruptions" in Mumford (1994), page 196, that invite an expansion $A'\supseteq A$ of A and a search for af-inertial (A',V)-models s' that are *dense paraphrases* (Ye et al., 2022) of s insofar as $\kappa_{A,V}(s')=s$. The trigger (16) for refining sigs can be extended to more elaborate constraints such as

$$\forall i\forall j\ (\delta_v(i,j)\supset\bigvee_{u\in A_v}(P_u(i)\wedge\chi^u(i,j))) \quad (17)$$

which conjoins $P_u(i)$ with a suitable description $\chi^u(i,j)$ of an *event nucleus* around the *culmination* u with a *preparatory process* at i and *consequent state* at j (Moens and Steedman, 1988). (17) reduces to (16) if χ^u is a tautology, but may otherwise take us outside (A,V), depending on how the preparatory process and consequent state are fleshed out. To keep the direction from state change to acts in (16), we can recast (17) as

$$\forall i\forall j\ ((P_u(i)\wedge iSj)\ \supset\chi^u(i,j)) \qquad (18)$$

for the reverse direction from acts to state change (and between (16) and (18), a cleaner interplay between A and V than in (17)).

For a concrete illustration, consider again

Facebook bought Instagram (4)

Facebook owns Instagram (5)

$$\boxed{\text{bought}(x,y)}\Rightarrow\boxed{\text{owns}(x,y)} \qquad (6)$$

The step from (4) to (5) suggested by the tensed predicates in (6) becomes more inviting if we insert *has* before *bought* in (4), and less so with *had*.

Facebook bought Instagram. $\boxed{E,R}\boxed{S}$ (19)

Facebk has bought Instagram. $\boxed{E}\boxed{R,S}$ (20)

Facebk had bought Instagram. $\boxed{E}\boxed{R}\boxed{S}$ (21)

Without R, the strings in (19) to (21) collapse to

$\boxed{E}\boxed{S}$ *Past*(*buy*(*facebook,instagram*))

The issue for (5) is: does the result *own*(*facebook, instagram*) of the *buy*(*facebook,instagram*)-event at E hold at the same box as S (assuming a sufficiently coarse notion of speech time so that S can serve both (4) and (5)). If *own*(*facebook,instagram*) coincides with R, the leap to (5) becomes easier from (20), if not from (19) or less, from (21). Expanding the sig (A,V), perhaps through (17), provides the ingredients for a more intricate account.

4 Conclusion

A triadic system (**Sig**, **Mod**, *Sen*) of finite-state representations is presented above, describing events and states through a vocabulary $(\mathsf{Act}, \mathsf{Val})$ of active and stative predicates. Finite fragments (A, V) of $(\mathsf{Act}, \mathsf{Val})$ are collected in **Sig**, from which **Mod** compresses strings of (A, V)-boxes, and *Sen* forms (A, V)-sentences defining sets of strings accepted by finite automata. As the compression on (A, V)-models can be computed by finite-state transducers, the (A, V)-sentences are closed under modal operators that turn the triad $(\mathbf{Sig}, \mathbf{Mod}, \mathit{Sen})$ into an institution. Four types of deformations that Mumford claims shape patterns at various levels of cognitive processing can be discerned in these semantic representations

(D1) blur in approximating $(\mathsf{Act}, \mathsf{Val})$ by (A, V)

(D2) domain warping from compressing strings for (A, V)-models of $\mathit{it}_{A,V}$*-from-*$\mathit{bit}_{A,V}$

(D3) superposition implemented over (A, V)-sentences representing sets of (A, V)-models

(D4) interruptions marked by (A, V, af)-accounts of inertia and cause-and-effect.

The deformations point to the brittleness of the semantic representations: (D1) to the limited detail in any sig (A, V); (D2) to the dependence of a model's domain (i.e., time) on its vocabulary; (D3) to the need to step from an (A, V)-model to an (A, V)-sentence; and (D4) to the step from an (A, V)-sentence to a range of (A', V')-sentences over various refinements (A', V') of (A, V).

The steps here are roughly comparable to Pearl's "ladder of causation" with rungs for observing, doing, and imagining (Pearl and Mackenzie, 2018). To say more, the obvious next step would be to bring in probabilities and noise. That anything at all could be said before taking that step reflects the extent to which causal graphs can be drawn and paths in them found without numbers (in line with Pearl (2009)'s *Causal-Statistical Dichotomy*).

Staying with what is presented above, let us return to the question with which we began: what does a string $\mathfrak{s}$ that is assigned a probability by a language model describe? We have focused on the case where $\mathfrak{s}$ is uttered to describe a particular event or situation, ignoring examples such as (22) that are not restricted to any particular situations, or (23) that are just one of many opinions.

Facebook spreads lies. (22)

Facebook is evil. (23)

To support a range of situations and views, increasingly complex structures are proposed above around an explicit notion of granularity, signature. A signature provides a handle on the variation supported, to keep matters from getting out of hand. Try as we might to get things right, however, the concluding lines in Reichenbach (1947) are telling.

> The history of language shows that logical categories were not clearly seen in the beginnings of language but were the results of long developments; we therefore should not be astonished if actual language does not always fit the schema which we try to construct in symbolic logic. A mathematical language can be coordinated to actual language only in the sense of an approximation.

Computational linguists have long complained about the brittleness of semantic representations; it is time for semanticists to own it. Our representations are brittle because, as approximations, they get bits wrong. But mistakes (which experience/data corrects) feed learning, which is what grammatical inference and pattern theory are about, not to mention the engine behind the astonishing technological strides of recent years. By comparison, the approximations Reichenbach refers to are corrected at a glacial, ponderous pace. Though that too is learning. The main thrust of the present paper is to show how deformations from pattern theory drive us to steps up in abstractness — from a finite vocabulary to an expansion of it, around which strings and their projections are (contravariantly) formed, and further up to sets of strings and their (covariant) refinements. Nor can we stop at any fixed institution, except for constraints of space and time that force these complications to be taken up elsewhere.

And so, while it may be difficult to pin down what, in general, a string assigned a probability by a language model describes, this much can be said. The string is about an open process, approximated (as far as we can tell) by representations of bounded but refinable granularity. Fleshing this out, the technicalities above represent an attempt to marry (if you will) the information-theoretic approach to pattern theory outlined in Mumford (1994) with institutions, understood according to Goguen (2006) as an elaboration of C.S. Peirce's triadic theory of signs, semiotics (and perhaps, process of signing, semiosis; e.g., Atkin, 2023).

References

N. Asher and A. Lascarides. 2003. *Logics of Conversation*. Cambridge University Press.

A. Atkin. 2023. Peirce's theory of signs. In E.N. Zalta and U. Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Stanford, https://plato.stanford.edu/archives/spr2023/entries/peirce-semiotics/.

J. Barwise and J. Perry. 1983. *Situations and Attitudes*. MIT Press.

E.M. Bender and A. Koller. 2020. Climbing towards NLU: on meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185 – 5198.

U. Coope. 2001. Why does Aristotle say that there is no time without change? *Proceedings of the Aristotelian Society*, 101:359–367.

R. Cooper. 2023. *From Perception to Communication: A Theory of Types for Action and Meaning*. Oxford University Press.

R. Cooper and J. Ginzburg. 2015. TTR for natural language semantics. In S. Lappin and C. Fox, editors, *Handbook of Contemporary Semantic Theory*, second edition, pages 375–407. Wiley-Blackwell.

D.R. Dowty. 1979. *Word Meaning and Montague Grammar*. Reidel.

I.A. Durand and S.R. Schwer. 2008. A tool for reasoning about qualitative temporal information: the theory of S-languages with a Lisp implementation. *J. Univers. Comput. Sci.*, 14(20):3282–3306.

T. Fernando. 2015. The semantics of tense and aspect: a finite-state perspective. In S. Lappin and C. Fox, editors, *Handbook of Contemporary Semantic Theory*, second edition, pages 203–236. Wiley-Blackwell.

T. Fernando. 2022. Strings from neurons to language. In *Proc. Natural Logic meets Machine Learning III*, pages 1–10. ACL Anthology.

M. Gelfond and V. Lifschitz. 1998. Action languages. *Linköping Electronic Articles in Computer and Information Science*, 3(16).

J.A. Goguen. 2006. Institutions and Peirce's triadic semiotics. https://cseweb.ucsd.edu/~goguen/projs/inst-sidebar.html. Accessed: 2023-04-21.

J.A. Goguen and R.M. Burstall. 1992. Institutions: Abstract model theory for specification and programming. *Journal of the ACM*, 39(1):95–146.

U. Grenander and M. Miller. 2007. *Pattern Theory: From Representation to Inference*. Oxford University Press.

D. Harel, D. Kozen, and J. Tiuryn. 2000. *Dynamic Logic*. MIT Press.

M.J. Hosseini. 2020. Unsupervised learning of relational entailment graphs from text. PhD Thesis, School of Informatics, University of Edinburgh.

M.J. Hosseini, S.B. Cohen, M. Johnson, and M. Steedman. 2019. Duality of link prediction and entailment graph induction. In *Proc 57th ACL*, pages 4736–4746. Florence, Italy.

H. Kamp and U. Reyle. 1993. *From Discourse to Logic*. Kluwer Academic Publishers.

A. Kehler. 2022. Coherence establishment as a source of explanation in linguistic theory. *Annual Review of Linguistics*, 8:123–42.

S.C. Kleene. 1956. Representation of events in nerve nets and finite automata. In C. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–41. Princeton University Press.

W. Klein. 2009. How time is encoded. In *The Expression of Time*, pages 39–82. De Gruyter Mouton.

P. Liang and C. Potts. 2015. Bringing machine learning and compositional semantics together. *Annual Review of Linguistics*, 1:355–376.

L. Libkin. 2004. *Elements of Finite Model Theory*. Springer.

M. Moens and M. Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15–28.

D. Mumford. 1994. Pattern Theory: a unifying perspective. In *First European Congress of Mathematics*, pages 187–224. Birkhäuser.

D. Mumford. 2019. A tribute to Ulf Grenander. *Quarterly of Applied Mathematics*, 77(2):201–206.

R. Nelken and N. Francez. 1995. Splitting the reference time: Temporal anaphora and quantification in DRT. In *Seventh Conference of the European Chapter of the Association for Computational Linguistics*.

C. K. Ogden and I. A. Richards. 1923. *The Meaning of Meaning: A Study of the Influence of Language upon Thought and of the Science of Symbolism*. Harcourt Brace Jovanovich, Inc.

J. Pearl. 2009. *Causality*, second edition. Cambridge University Press.

J. Pearl and D. Mackenzie. 2018. *The Book of Why*. Penguin.

W.V.O. Quine. 1950. *Methods of Logic*. Henry Holt and Co.

H. Reichenbach. 1947. The tenses of verbs. In *Elements of Symbolic Logic*, pages 287–298. Macmillan & Co.

D. Sannella and A. Tarlecki. 2015. The foundational legacy of ASL. In *Software, Services and Systems: Essays Dedicated to Martin Wirsing on the Occasion of His Retirement from the Chair of Programming and Software Engineering*, volume 8950 of LNCS, pages 253–272. Springer.

J.A. Wheeler. 1990. Information, physics, quantum: The search for links. In W. Zurek, editor, *Complexity, Entropy and the Physics of Information*, pages 3–28. Addison-Wesley.

B. Ye, J. Tu, E. Jezek, and J. Pustejovsky. 2022. Interpreting logical metonymy through dense paraphrasing. In *Proc 44th Annual Conference of the Cognitive Science Society*, pages 3541–3548.