Strings from Neurons to Language

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

Abstract

Chains of transitions by finite automata originally conceived to analyze neural events are described at different granularities by strings. The granularities are refined, and transformed into increasingly elaborate structures, against which to understand the events recorded in the strings. Choosing the correct structure is a problem of induction and learning. The events and strings studied arise in natural language semantics.

1 Introduction

Natural language inference concerns connections that may or may not exist between say, (i) and (ii).

(i) Facebook bought Instagram.

(ii) Facebook owns Instagram.

If (ii) follows from (i), we might add the link

$$\boxed{\text{facebook}} \xrightarrow{\text{owns}} \boxed{\text{instagram}}$$

to a knowledge graph which already has the link

$$\boxed{\text{facebook}} \xrightarrow{\text{bought}} \boxed{\text{instagram}}.$$ [1]

But does (ii) follow from (i)? What if an event happened after the purchase (i) transferring ownership of Instagram away from Facebook (perhaps to Meta)? Rather than insisting that buy(X,Y) entails own(X,Y) by leaving out such troublesome events, the present paper proposes that buy(X,Y) entails *Become*(own(X,Y))

$$\boxed{\text{buy(X,Y)}} \Rightarrow \boxed{\textit{Become}\text{(own(X,Y))}} \qquad (1)$$

as pictured by a transition

$$\boxed{\neg\text{own(X,Y)}} \xrightarrow{\text{buy(X,Y)}} \boxed{\text{own(X,Y)}} \qquad (2)$$

(associating the precondition ¬own(X,Y) and postcondition own(X,Y) with the act buy(X,Y)) which may (or may not) follow (or precede) a transition such as

$$\boxed{\text{own(X,Y)}} \xrightarrow{\text{sell(X,Y)}} \boxed{\neg\text{own(X,Y)}}$$

(swapping the preconditions and postconditions in (2)) to describe further changes in ownership. The operator *Become* in (1) can be found in the aspectual calculus of Dowty (1979) and characterized by entailments

$$\boxed{\textit{Become}\text{(A)}}\boxed{} \Rightarrow \boxed{\neg\text{A}}\boxed{\text{A}} \qquad (3)$$

and

$$\boxed{\neg\text{A}}\boxed{\text{A}} \Rightarrow \boxed{\textit{Become}\text{(A)}}\boxed{} \qquad (4)$$

using the same binary connective $\Rightarrow$ in (1) to map regular languages L and L' to a regular language $L \Rightarrow L'$ (see, for example, §3.4 of Fernando (2015)).[2]

Whatever semantics is (or is not) attached to $\Rightarrow$ in (1), it is clear that there is more to a *buy*-event than the change in ownership expressed in (1),(2); no mention is made, for instance, of a payment that is part of any *buy*-event. While this omission does not diminish the entailment (1), it suggests there is more to the pre-states and post-states of a buy(X,Y)-transition than is on display in the boxes

$$\boxed{\neg\text{own(X,Y)}} \quad \text{and} \quad \boxed{\text{own(X,Y)}}$$

[1] A recent work on link prediction and entailment graphs is Hosseini et al. (2019).

[2] That is, (1), (3) and (4) are not unlike constraints in finite-state morphology (e.g., Beesley and Karttunen, 2003), except that the symbols constituting the alphabet of the languages for $\Rightarrow$ are assumed throughout to be sets. These sets are drawn with boxes (rather than the customary curly braces $\{,\}$ and $\emptyset$) to distinguish sets *qua* symbols (as in the string $\boxed{}$ of length 1) from sets *qua* languages (e.g., the language $\emptyset$ without any strings, not to mention the empty string ϵ of length 0).

in (2). To salvage (2), let us bring out the (bounded) granularity Σ underpinning (2), and assert that if

$$q \stackrel{\text{buy(X,Y)}}{\longrightarrow} q'$$

then the states q and q' are Σ-approximated by

$$\boxed{\neg\text{own(X,Y)}} \quad \text{and} \quad \boxed{\text{own(X,Y)}}$$

respectively. That is, (2) becomes

$$q \stackrel{\text{buy(X,Y)}}{\longrightarrow} q' \quad \text{with } \pi_\Sigma(q) = \boxed{\neg\text{own(X,Y)}} \\ \text{and } \pi_\Sigma(q') = \boxed{\text{own(X,Y)}} \tag{5}$$

where π_Σ maps a state to its Σ-approximation. But what exactly is this granularity Σ and map π_Σ? And how can we refine Σ to establish an entailment

$$\boxed{\text{buy(X,Y)}} \Rightarrow \boxed{\text{pay(X,Y)}} \tag{6}$$

injecting an ingredient, pay(X,Y), missing from (1),(2)?

To answer these and related questions, the present work defines three notions, a *transition signature* Σ, a Σ*-strip* and an X*-projection relative to* Σ, under which a chain

$$q_0 \stackrel{a_1}{\to} q_1 \stackrel{a_2}{\to} q_2 \stackrel{a_3}{\to} \cdots \stackrel{a_n}{\to} q_n \tag{7}$$

of transitions $q_i \stackrel{a_{i+1}}{\to} q_{i+1}$ from state q_i to state q_{i+1} over a_{i+1} can be formulated as strings of varying granularities, capturing finite fragments of q_i and of a_i. The somewhat surprising suggestion here is that there are strings other than $a_1 a_2 \cdots a_n$ to extract from the chain (7), and proper fragments of q_i and of a_i to describe. This suggestion becomes less surprising when we turn to the source (Kleene, 1956) of finite automata; in the application there to nerve nets, a_i is a set and q_i is a record. This is explained in section 2, where transition signatures Σ and Σ-strips are defined, under which the transition (2) can be encoded as the string

$$\boxed{\text{(own(X,Y),0), buy(X,Y)}}\boxed{\text{(own(X,Y),1)}}$$

of length 2; its first symbol is the box consisting of the act buy(X,Y) and the ordered pair (own(X,Y),0) saying own(X,Y) is 0/false; its second symbol is the box consisting of the single ordered pair (own(X,Y),1) saying own(X,Y) is 1/true. More generally, (7) becomes a string $\alpha_1\alpha_2 \cdots \alpha_k$ of boxes α_i formed by adding acts to records, or better still (when varying a signature within a category), record types (applied to linguistic semantics in Cooper and Ginzburg (2015)). As for the entailment (6), this is addressed through X-projections (relative to Σ), defined in section 3, where Russell-Wiener event structures (Kamp and Reyle, 1993, pages 667–674) and interval relations from Allen (1983) are revisited. Section 4 outlines how to deploy the three notions defined within logical settings for learning finite automata from strings associated with signatures. The claim is that the step from strings to automata tracks the move from episodic reports (such as (i), *Facebook bought Instagram*) to generic statements such as

(iii) Facebook spreads lies.

The ideas described below, including the connection to neural nets, are intended to make this claim plausible[3] and intriguing.

2 Nerve nets and beyond

Finite automata go back to Kleene (1956)'s analysis of a nerve net from McCulloch and Pitts (1943) consisting of finite numbers k and m of

(i) *input cells*, $\mathcal{N}_1, \ldots, \mathcal{N}_k$, described at different times by different symbols from a finite alphabet A, and

(ii) *inner cells*, $\mathcal{M}_1, \ldots, \mathcal{M}_m$, described at different times by different states from a finite set Q.

In Rabin and Scott (1959), (i) and (ii) are put aside in favor of a "black box" perspective on finite automata, moving them away from nerve nets. Widely adopted in textbook accounts of finite automata, this perspective has proved enormously fruitful. It has, however, also resulted in some ideas from Kleene (1956) being sidelined, including the possibility from (i) and (ii) above that

(†) a state q is an m-tuple $(v_1, \ldots, v_m)$ and a transition $q' \stackrel{a}{\to} q$ combines m simpler relations $\to_1, \ldots, \to_m$

$$q' \stackrel{a}{\to} q \quad \text{iff} \quad q' \stackrel{a}{\to}_i v_i \text{ for all } i \in \{1, \ldots, m\}$$

where $q' \stackrel{a}{\to}_i v_i$ depends on only certain parts of q' and of a (for $1 \leq i \leq m$).

[3]Generics have been linked to causation (e.g., Carlson, 1995); automata are obvious candidates for causal structures.

The m in (†) is the same number m in (ii) of inner cells $\mathcal{M}_1, \ldots, \mathcal{M}_m$; the state q in (†) assigns values $v_1, \ldots, v_m$ to $\mathcal{M}_1, \ldots, \mathcal{M}_m$, respectively. Apart from k and m, Kleene (1956) assumes each inner cell $\mathcal{M}_i$ can be assigned any of $s_i \geq 2$ different values (of which v_i is one), leading to $\prod_{i=1}^{m} s_i$ many m-tuples $(v_1, \ldots, v_m)$ in the state set Q. The all-or-none assumption of nervous activity in McCulloch and Pitts (1943) is applied to the k input cells in (i) for an alphabet A consisting of the 2^k subsets a of $\{\mathcal{N}_1, \ldots, \mathcal{N}_k\}$, the intention being that a describes a time where an input cell is active (firing) if and only if it is in a. The transition $q' \xrightarrow{a}_i v_i$ in (†) is subject to activation laws specifying how to update the value of $\mathcal{M}_i$ when the inner cells have values q' and a is the set of active input cells.[4] If $q' = (v'_1, \ldots, v'_m)$, then $q' \xrightarrow{a}_i v_i$ does not depend on any v'_j describing an inner cell $\mathcal{M}_j$ that does not feed into $\mathcal{M}_i$ nor on any input cell in a that does not feed into $\mathcal{M}_i$. (The nerve nets may or may not be fully recurrent.)

Extracting the string $a_1 a_2 \cdots a_n$ from the chain

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \cdots \xrightarrow{a_n} q_n \qquad (7)$$

of transitions leaves out the states $q_0, q_1, \ldots, q_n$, which in Kleene (1956) describe m inner cells at $n+1$ times. This reflects a focus on the external environment that is connected to inner cells via inputs cells (described at n times by $a_1 a_2 \cdots a_n$). Away from the particularities of nerve nets, however, no such separation between external and internal matters need keep us from extracting instead the string $q_0 q_1 \cdots q_n$ from (7). In line with Dowty (1979)'s use of stative predicates as the basis for his aspectual calculus, we apply strings $q_0 q_1 \cdots q_n$ in section 3 to represent the finitely many events mentioned in a (finite) discourse. Some middle ground between strings $q_0 q_1 \cdots q_n$ of states and the usual strings $a_1 a_2 \cdots a_n \in A^*$ is staked out by strings $\alpha_0 \alpha_1 \cdots \alpha_n$ of finite sets α_i that provide information about states q_i and symbols a_{i+1} (for $i < n$) alike. That information may, in (†) above, zero in on the parts of q' and of a on which $q' \xrightarrow{a}_i p_i$ depends.

Two basic intuitions shape the work on strings $\alpha_0 \alpha_1 \cdots \alpha_n$ below. The first is that

($*$) a string which represents a chain (7) of transitions is a data point that is to be explained (alongside other data points) with more complex structures

and the second is that

($**$) to keep the structures in ($*$) managable, we associate a string $\alpha_0 \alpha_1 \cdots \alpha_n$ with a finite granularity which can be refined as information about it accumulates.

[4]These involve thesholds and two types of connections, inhibitory and excitatory (McCulloch and Pitts, 1943), or in the case of perceptrons, weights, biases and activation functions.

Kleene (1956)	transition signature
inner cell $\mathcal{M}_i$	label $\in \mathbf{L}$
$(s_1, \ldots, s_m)$	value-sets $\{\mathcal{V}(l)\}_{l \in \mathbf{L}}$
input cell $\mathcal{N}_i$	act $\in \mathbf{Act}$
neural connections	$\mathsf{af} : \mathbf{Act} \to 2^{\mathbf{L}}$

Table 1: Transition signatures in Kleene (1956)

Mention of data in ($*$) calls for a reference to *grammatical inference* (e.g., Heinz and Sempere, 2016; de la Higuera, 2010). The examples considered here, however, are from natural language semantics. Under ($*$), the step from accounts of events that happen (in actuality) up to general statements (including causal claims, counterfactuals and potentiality) is an inductive generalization over strings which demands richer structures. As for ($**$), the main thrust of the present paper is to formulate granularities as signatures (or vocabularies) familiar in model theory, preparing the ground for logical systems based on signatures called *institutions* (Goguen and Burstall, 1992). The finite signatures formed below keep the structures finite-state (connected in a precise sense with Kleene (1956)), making significant bits of the reasoning decidable (a theme from Rabin and Scott (1959)).

Getting down to business, let us package key aspects of Kleene (1956) in a signature, following Table 1 above.

Definition 1. A *transition signature* is a 4-tuple $\Sigma = (\mathbf{L}, \mathcal{V}, \mathbf{Act}, \mathsf{af})$, where

(i) $\mathbf{L}$ is a finite set of labels,

(ii) $\mathcal{V}$ is a function with domain $\mathbf{L}$ assigning each label l a finite set $\mathcal{V}(l)$ of l-values,

(iii) $\mathbf{Act}$ is a finite set of acts distinct from pairs (l, v) of labels l and l-values v

$$\mathbf{Act} \cap \{(l, v) \mid l \in \mathbf{L} \text{ and } v \in \mathcal{V}(l)\} = \emptyset$$

and

(iv) $\mathsf{af} : \mathbf{Act} \to 2^{\mathbf{L}}$ is a function specifying the set $\mathsf{af}(a)$ of labels that an act a can affect.

In Kleene (1956), labels are inner cells, acts are input cells, and af maps every input cell to the set of inner cells it is connected to. For the transition

$$\boxed{\neg\text{own(X,Y)}} \overset{\text{buy(X,Y)}}{\longrightarrow} \boxed{\text{own(X,Y)}} \qquad (2)$$

(from the previous section), let

$$\Sigma \;=\; (\{\text{own(X,Y)}\},\ \mathcal{V},\ \{\text{buy(X,Y)}\},\ \mathsf{af})$$

where

$$\mathcal{V}(\text{own(X,Y)}) = \{0, 1\}$$

and

$$\mathsf{af}(\text{buy(X,Y)}) = \{\text{own(X,Y)}\}$$

to encode (2) as the string

$$\boxed{\text{(own(X,Y),0), buy(X,Y)}}\boxed{\text{(own(X,Y),1)}}.$$

In general, a transition signature Σ has a *stative part* $Q(\Sigma)$ equal to the set of $\mathcal{V}$-records, where a *$\mathcal{V}$-record* is a function q with domain $\mathbf{L}$ mapping each $l \in \mathbf{L}$ to an l-value $q(l) \in \mathcal{V}(l)$. The disjointness in clause (iii) of Definition 1 prevents any confusion when forming a string $\alpha_0\alpha_1 \cdots \alpha_n$ of subsets α_i of

$$\mathbf{Act} \cup \{(l, v) \mid l \in \mathbf{L} \text{ and } v \in \mathcal{V}(l)\}$$

to specify a chain

$$q_0 \overset{a_1}{\to} q_1 \overset{a_2}{\to} q_2 \overset{a_3}{\to} \cdots \overset{a_n}{\to} q_n \qquad (7)$$

of transitions where q_i is the part of α_i with*out* acts

$$q_i := \alpha_i \setminus \mathbf{Act} \quad \text{(i.e., } \{a \in \alpha_i \mid a \notin \mathbf{Act}\})$$

and a_{i+1} is the subset of α_i consisting of acts

$$a_{i+1} := \alpha_i \cap \mathbf{Act} \quad (\text{for } i < n).$$

We can sidestep the disjointness requirement by turning each set α_i in $\alpha_0\alpha_1 \cdots \alpha_n$ into an ordered pair (q_i, a_{i+1}) of a $\mathcal{V}$-record q_i and subset a_{i+1} of $\mathbf{Act}$; we opt here instead for the union

$$q_i \cup a_{i+1} = \alpha_i.$$

Let us define a *Σ-box* α to be the union of a $\mathcal{V}$-record with a subset of $\mathbf{Act}$. Given a Σ-box α and a label $l \in \mathbf{L}$, let us agree that the *value of l at α* is the unique l-value v such that $(l, v) \in \alpha$. We say α and α' are *l-equivalent* and write $\alpha =_l \alpha'$ if l has the same value at α and α'. To express the idea that adjacent boxes in a string are l-equivalent unless the boxes are linked by an act affecting l, let $\overline{\mathsf{af}} : 2^{\mathbf{Act}} \to 2^{\mathbf{L}}$ be the function mapping each $\alpha \subseteq \mathbf{Act}$ to the set

$$\overline{\mathsf{af}}(\alpha) := \mathbf{L} \setminus \bigcup_{a \in \alpha} \mathsf{af}(a)$$

of labels *not* in $\mathsf{af}(a)$ for any $a \in \alpha$. For example,

$$\overline{\mathsf{af}}(\emptyset) = \mathbf{L}$$

as there is no act in $\emptyset$ to affect a label. A label is said to be *unaffected by* α if it belongs to $\overline{\mathsf{af}}(\alpha)$. Next we define strings basic to this paper.

Definition 2. Given a transition signature $\Sigma = (\mathbf{L}, \mathcal{V}, \mathbf{Act}, \mathsf{af})$, a *$\Sigma$-strip* is a string $\alpha_1 \cdots \alpha_n$ of Σ-boxes α_i such that $\alpha_n \cap \mathbf{Act} = \emptyset$ and for all i such that $1 \leq i < n$, $\alpha_i \cap \mathbf{Act} \neq \emptyset$ and

$$\alpha_i =_l \alpha_{i+1} \quad \text{for each } l \in \overline{\mathsf{af}}(\alpha_i \cap \mathbf{Act}). \qquad (8)$$

Line (8) in Definition 2 comes with a slogan

no change without force

on the understanding that $\alpha_i =_l \alpha_{i+1}$ means "no change" and that $\mathbf{Act}$ covers all relevant forces. (8) gives us a handle on change and the tendency to infer (ii) from (i) in the absence of any act affecting own(facebook,instagram) after a buy(facebook,instagram)-event.

(i) Facebook bought Instagram.

(ii) Facebook owns Instagram.

More on af and on what it says about refinements of Σ in the next section.

3 Events from intervals to strings

"An important part" of interpreting "a piece of discourse" is representing the "comparatively few events" mentioned in it, according to Kamp (2013). An event e is assumed in Allen (1983) and Kamp and Reyle (1993) to stretch over a temporal interval, leaving times before and after e. Under this assumption, a set E of events induces a notion of time as follows. Let us define an *E-state* $q = (U, A, D)$ to be a triple of subsets U, A, D of E that are pairwise disjoint and cover E

$$U \cap A = \emptyset \text{ and } D = E \setminus (U \cup A).$$

The idea is that (U, A, D) describes a time that is

(i) before every event in U (making U the set of *u*nborn events in E),

(ii) during every event in A (making A the set of *a*live events in E), and

(iii) after every event in D (making D the set of *d*ead events in E).

To capture the order implicit in this idea, we let Q_E be the set of E-states, and we represent the passage of time by a binary relation $\rightarrow_E$ on Q_E such that

$$(U, A, D) \rightarrow_E (U', A', D')$$

means

$$U' \subseteq U \text{ and } A \neq A' \text{ and } D \subseteq D' \subseteq D \cup A. \quad (9)$$

(9) says unborn events were in the past unborn ($U' \subseteq U$), the set of alive events changes ($A \neq A'$), and dead events stay dead, having at the previous moment been alive or dead ($D \subseteq D' \subseteq D \cup A$). To associate a transition signature $\Sigma = (\mathbf{L}, \mathcal{V}, \mathbf{Act}, \mathsf{af})$ with $\rightarrow_E$, we let $\mathbf{L} = E$, and fix a set $\{$u,a,d$\}$ of three values to which $\mathcal{V}$ maps every event in E, identifying an E-state $q = (U, A, D)$ with the function $\hat{q} : E \rightarrow \{u,a,d\}$ mapping $e \in E$ according to which of U, A, D has e

$$\hat{q}(e) := \begin{cases} \mathrm{u} & \text{if } e \in U \\ \mathrm{a} & \text{if } e \in A \\ \mathrm{d} & \text{otherwise (i.e., } e \in D). \end{cases}$$

The three e-values (u,a,d) are more than the two (0,1) needed by a label l such as own(facebook,instagram) to say l is true or false. For transitions such as $\rightarrow_E$ that do not specify any acts, we can express that non-specification through an anonymous act ☺ that can affect any of the labels. Putting

$$\mathbf{Act} = \{☺\} \text{ and } \mathsf{af}(☺) = \mathbf{L}$$

completes the $\rightarrow_E$-column of Table 2. An alternative to ☺ is to associate every event $e \in E$ with a left border l_e and a right border r_e for a set

$$E^{l,r} := \{l_e \mid e \in E\} \cup \{r_e \mid e \in E\}$$

of borders of E.[5] From the definition (9) of $(U, A, D) \rightarrow_E (U', A', D')$ above, we can then extract a non-empty subset

$$\{l_e \mid e \in A' \cap U\} \cup \{r_e \mid e \in D' \cap A\} \quad (10)$$

Σ	$\rightarrow_E$	actions (10)	synthesis
$\mathbf{L}$	E	$\emptyset$	E
$\mathcal{V}$	$\lambda e.\{\mathrm{u,a,d}\}$	$\emptyset$	$\lambda e.\{\mathrm{u,a,d}\}$
$\mathbf{Act}$	$\{☺\}$	$E^{l,r}$	$E^{l,r}$
af	$\{(☺, E)\}$	$\lambda a.\emptyset$	af_E

Table 2: Transition signatures for E as interval-strings

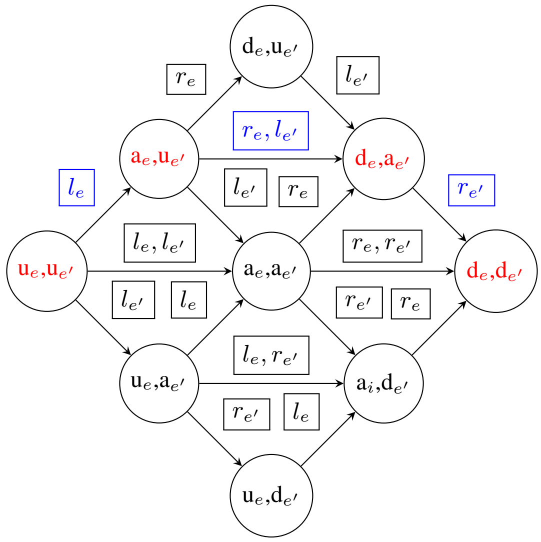

Figure 1: The relation $\rightarrow_{\{e,e'\}}$ labelled by actions (10)

of $E^{l,r}$, to express the transition from (U, A, D) to (U', A', D'). We can also use an event e as a subscript on the value an E-state q associates with e, under the repackaging $\hat{q} = \{v_e \mid e \in E\}$ where v_e abbreviates the pair (e, v) in $\hat{q}$. For instance, for $E = \{e, e'\}$, we can shorten the E-state $(E, \emptyset, \emptyset)$ to $\{\mathrm{u}_e, \mathrm{u}_{e'}\}$, the E-state $(\{e'\}, \{e\}, \emptyset)$ to $\{\mathrm{a}_e, \mathrm{u}_{e'}\}$, the E-state $(\emptyset, \{e'\}, \{e\})$ to $\{\mathrm{d}_e, \mathrm{a}_{e'}\}$, and the E-state $(\emptyset, \emptyset, E)$ to $\{\mathrm{d}_e, \mathrm{d}_{e'}\}$. These four E-states appear in red in Figure 1, with the sets (10) as boxes over arrows given by $\rightarrow_E$. The three blue boxes in Figure 1 form the string

$$\boxed{l_e}\boxed{r_e, l_{e'}}\boxed{r_{e'}} \quad (11)$$

corresponding to the Allen interval relation e *meets* e' (called abutment in Kamp and Reyle (1993)). All 13 interval relations in Allen (1983) are expressed in Figure 1 as strings labelling transitions from $\{\mathrm{u}_e, \mathrm{u}_{e'}\}$ to $\{\mathrm{d}_e, \mathrm{d}_{e'}\}$. The 13 strings over the 8 symbols $\boxed{l_e}$, $\boxed{l_e, l_{e'}}$, $\boxed{l_e, r_{e'}}$, $\boxed{r_e}$, $\boxed{r_e, l_{e'}}$, $\boxed{r_e, r_{e'}}$, $\boxed{l_{e'}}$, $\boxed{r_{e'}}$ appear in Durand and Schwer (2008) with*out* E-states. The derivation (10) of l_e and r_e from E-states supports the intuition de-

[5] An event e is, as it were, born with the injunction **l**ive, l_e, and dies with the injunction **r**est, r_e.

fended in Allen (1983) that intervals are conceptually prior to points such as l_e and r_e.

Indeed, we can construe l_e as *Become*(a_e) and r_e as *Become*(d_e), where *Become* is one of the "three or four sentential operators and connectives" through which David Dowty explains "the different aspectual properties of the various kinds of verbs" on the basis of "a single homogeneous class of predicates — stative predicates" (Dowty, 1979, page 71). The pairs a_e and d_e in *Become*(a_e) and *Become*(d_e) are stative insofar as they make up an E-state $\hat{q}$, changes to which are trigerred by *actions* made up of l_e and r_e.

Strings of actions such as

$$\boxed{l_e}\boxed{r_e, l_{e'}}\boxed{r_{e'}} \tag{11}$$

differ from strings of E-states such as

$$\boxed{\mathrm{u}_e, \mathrm{u}_{e'}}\boxed{\mathrm{a}_e, \mathrm{u}_{e'}}\boxed{\mathrm{d}_e, \mathrm{a}_{e'}}\boxed{\mathrm{d}_e, \mathrm{d}_{e'}} \tag{12}$$

(red in Figure 1) in an important respect that is revealed when reducing the set E of events to a smaller set. For this, a definition is helpful. Given a set X and a string $s = \alpha_1 \cdots \alpha_n$ of sets α_i, the *X-reduct $\rho_X(s)$ of s* is s intersected componentwise with X

$$\rho_X(\alpha_1 \cdots \alpha_n) \ := \ (\alpha_1 \cap X) \cdots (\alpha_n \cap X)$$

(Fernando, 2015). For example, the $\{l_e, r_e\}$-reduct of (11) is

$$\rho_{\{l_e, r_e\}}(\boxed{l_e}\boxed{r_e, l_{e'}}\boxed{r_{e'}}) \ = \ \boxed{l_e}\boxed{r_e}\boxed{} \tag{13}$$

while the $\{\mathrm{u}_e, \mathrm{a}_e, \mathrm{d}_e\}$-reduct of (12) is

$$\boxed{\mathrm{u}_e}\boxed{\mathrm{a}_e}\boxed{\mathrm{d}_e}\boxed{\mathrm{d}_e}. \tag{14}$$

Strings (13) and (14) can be extracted from the chain

$$\boxed{\mathrm{u}_e} \xrightarrow{\boxed{l_e}} \boxed{\mathrm{a}_e} \xrightarrow{\boxed{r_e}} \boxed{\mathrm{d}_e} \xrightarrow{\boxed{}} \boxed{\mathrm{d}_e} \tag{15}$$

of transitions, which we can truncate to

$$\boxed{\mathrm{u}_e} \xrightarrow{\boxed{l_e}} \boxed{\mathrm{a}_e} \xrightarrow{\boxed{r_e}} \boxed{\mathrm{d}_e} \tag{16}$$

in accordance with the Aristotelian dictum

no time without change

where change is observed through the elements of X. Truncating (15) to (16) removes the empty box $\square$ in (13) and the stutter $\boxed{\mathrm{d}_e}\boxed{\mathrm{d}_e}$ in (14). This suggests forming the X-projection of a string s by compressing its X-reduct $\rho_X(s)$; that compression is Durand and Schwer (2008)'s deletion $d^\square$ of $\square$

$$\begin{aligned} d^\square(\epsilon) \ &:= \epsilon \quad \text{(empty string)} \\ d^\square(\alpha s) \ &:= \begin{cases} d^\square(s) & \text{if } \alpha = \square \\ \alpha d^\square(s) & \text{otherwise} \end{cases} \end{aligned}$$

and Fernando (2015)'s elimination bc of stutters

$$\begin{aligned} bc(s) \ &:= s \qquad \text{if length}(s) < 2 \\ bc(\alpha\alpha' s) \ &:= \begin{cases} bc^A(\alpha' s) & \text{if } \alpha = \alpha' \\ \alpha bc^A(\alpha' s) & \text{otherwise.} \end{cases} \end{aligned}$$

Returning to the transition signatures in Table 1, the two middle columns (in blue and red) agree in allowing every act to affect every label

$$\mathsf{af}(a) = \mathbf{L} \quad \text{for every } a \in \mathbf{Act} \tag{17}$$

(as ☺ is the only act in the $\rightarrow_E$-column, and there are no labels in the column next to it). For the fourth component af of a signature Σ to do any work (i.e., for line (8) in Definition 2 to be non-vacuous), neither its stative part $Q(\Sigma)$ nor its active part $\mathbf{Act}$ should be trivial. This brings us to the rightmost column of Table 1, where the specificity of the acts l_e and r_e is captured by the equation

$$\mathsf{af}_E(l_e) = \{e\} = \mathsf{af}_E(r_e) \quad \text{for } e \in E$$

which, if $|E| = 1$, reduces to (17) but is quite different otherwise.

The question arises: how do we define the X-projection of a string s of sets with non-trivial stative and non-stative parts? We compress its X-reduct $\rho_X(s)$ by splitting X between its intersections with $\mathbf{Act}$ and with the complement of $\mathbf{Act}$

$$A = X \cap \mathbf{Act} \ \text{ and } \ B = X \setminus \mathbf{Act}.$$

In case $B = \emptyset$, we remove all occurrences of the empty box from $\rho_A(s)$ for $d^\square(\rho_A(s))$. Otherwise, we eliminate stutters $\alpha\alpha$ whenever α does not intersect A, as carried out by κ^A

$$\begin{aligned} \kappa^A(s) \ &:= s \qquad \text{if length}(s) < 2 \\ \kappa^A(\alpha\alpha' s) \ &:= \begin{cases} \kappa^A(\alpha' s) & \text{if } \alpha = \alpha' \text{ and} \\ & \quad \alpha \cap A = \emptyset \\ \alpha\kappa^A(\alpha' s) & \text{otherwise} \end{cases} \end{aligned}$$

(so that κ is just $\kappa^{\emptyset}$). Putting these two cases together, let the (A, B)*-projection* $\kappa_{A,B}(s)$ *of* s be

$$\kappa_{A,B}(s) \ := \begin{cases} d^{\square}(\rho_A(s)) & \text{if } B = \emptyset \\ \kappa^A(\rho_{A\cup B}(s)) & \text{otherwise.} \end{cases}$$

For the record, we have

Definition 3. Given a set X and a string s of sets, the X*-projection of* s *relative to* a transition signature $\Sigma = (\mathbf{L}, \mathcal{V}, \mathbf{Act}, \mathsf{af})$ is the (A, B)-projection $\kappa_{A,B}(s)$, where A is $X \cap \mathbf{Act}$ and B is $X \setminus \mathbf{Act}$. When it is clear what Σ is, we shorten $\kappa_{A,B}(s)$ to $\kappa_X(s)$ and refer to it simply as the X*-projection of* s.

If $\hat{s}$ is the string

$$\boxed{\mathrm{u}_e, \mathrm{u}_{e'}, l_e}\boxed{\mathrm{a}_e, \mathrm{u}_{e'}, r_e, l_{e'}}\boxed{\mathrm{d}_e, \mathrm{a}_{e'}, r_{e'}}\boxed{\mathrm{d}_e, \mathrm{d}_{e'}}$$

then $\kappa_{\{l_e, r_e, \mathrm{u}_e, \mathrm{a}_e, \mathrm{d}_e\}}(\hat{s})$ is the string

$$\boxed{\mathrm{u}_e, l_e}\boxed{\mathrm{a}_e, r_e}\boxed{\mathrm{d}_e}$$

depicting the chain (16) above. In this particular case, the X-projection of a Σ-strip is a Σ_X-strip, where Σ_X is the transition signature that X reduces Σ to. In general, however, the X-projection of a Σ-strip need not be a Σ_X-strip. Such projections can be viewed as disfavored models, where we might find counterexamples to *Facebook owns Instagram* even though *Facebook bought Instagram*.

Let us summarize this section. Transitions $\to_E$ based on a set E of events-as-intervals are strung out to Σ-strips, turning E into a set of labels of records, boxed alongside acts. The sortal distinction between acts and statives (built into the transition signature Σ) is applied to the compression of X-reducts, yielding X-projections at a granularity coarser than Σ.[6]

4 Finite-state elaborations

Definitions 1-3 from sections 2 and 3 above are part of an attempt to work out (over strings) a basic aspectual difference between *buy* and *own*, glossed over by links

$$\boxed{\text{facebook}} \xrightarrow{\text{owns}} \boxed{\text{instagram}} \tag{18}$$

and

$$\boxed{\text{facebook}} \xrightarrow{\text{bought}} \boxed{\text{instagram}} \tag{19}$$

[6]With the stative/non-stative distinction in place, events as intervals can be refined to Vendler classes (e.g., Moens and Steedman, 1988; Fernando, 2020).

(from a knowledge graph), but represented by a transition

$$\boxed{\neg\text{own(X,Y)}} \xrightarrow{\text{buy(X,Y)}} \boxed{\text{own(X,Y)}} \tag{2}$$

(in a finite automaton). The transition (2) suggests that inferring (18) from (19) is (to put it gently) complicated. But if we are to take the transition (2) seriously as a tool for lexical semantics, we must acknowledge too that buy(X,Y) is more complicated than (2), involving, as it does, acts such as pay(X,Y) left out of (2). Accordingly, we take pains to associate a certain transition signature $\Sigma_\circ$ with (2), which we encode as the $\Sigma_\circ$-strip

$$\boxed{\text{(own(X,Y),0), buy(X,Y)}}\boxed{\text{(own(X,Y),1)}} \tag{20}$$

(see section 2). This $\Sigma_\circ$-strip can be obtained from transition signatures with larger vocabularies through X-projections, where X is the set

$$\{\text{(own(X,Y),0), buy(X,Y), (own(X,Y),1)}\}$$

from which the boxes in (20) are formed (see section 3). In particular, we may add an act pay(X,Y) to the transition signature $\Sigma_\circ$ for a more refined transition signature on which to impose the entailment

$$\boxed{\text{buy(X,Y)}} \Rightarrow \boxed{\text{pay(X,Y)}} \tag{6}$$

adding pay(X,Y) to the first box in (20) because that box has buy(X,Y). Fleshing out the preconditions and postconditions of pay(X,Y) may require further expansions of the transition signature's vocabulary. Each expansion is finite and is (with any luck) not the last, reflecting the open-endedness of events described in natural language. Refining Σ may not only fill boxes in a Σ-strip; it may also lengthen the Σ-strip, as one transition follows another. This is why we consider chains of more than a single transition, and why (i) does *not* entail (ii).

(i) Facebook bought Instagram.

(ii) Facebook owns Instagram.

The increase in string length is turned into a decrease when, in section 3, the X-reduct $\rho_X(s)$ of a string s is compressed to form its X-projection $\kappa_X(s)$ (relative to a signature distinguising acts from the label-value pairs of records). This is because a projection moves to a coarser granularity, rather than (as in the case of an embedding) a finer one. More precisely, given a category **Sign** of signatures where a morphism $\sigma : \Sigma \to \Sigma'$ embeds a signature, Σ, into a finer one, Σ', a functor **Mod** that is *contra*variant on **Sign** returns a projection $\mathbf{Mod}(\sigma)$ coarsening $\mathbf{Mod}(\Sigma')$ down to $\mathbf{Mod}(\Sigma)$. The category **Sign** and functor **Mod** constitute part of a logical system

$$(\mathbf{Sign}, \mathbf{Mod}, \mathit{Sen}, \{\models_\Sigma\}_{\Sigma \in |\mathbf{Sign}|})$$

called an *institution* (Goguen and Burstall, 1992) in which

(i) the functor **Mod** maps Σ contravariantly to a category $\mathbf{Mod}(\Sigma)$ of Σ-models,

(ii) a covariant functor Sen maps Σ to a set $\mathit{Sen}(\Sigma)$ of Σ-sentences, and

(iii) for each signature Σ, $\models_\Sigma$ is a binary relation between Σ-models and Σ-sentences that meets a certain *Satisfaction Condition* discussed below.

But how is a string of sets to be understood as a model of predicate logic? For any set U and string $s = \alpha_1\alpha_2 \cdots \alpha_n$ of subsets α_i of U, let $M_U[s]$ be the U*-structure*

$$M_U[s] \ = \ ([n], <_n, \{[\![P_u]\!]\}_{u \in U})$$

over a universe $[n] \ = \ \{1, 2, \ldots, n\}$ of string positions with $<_n$ as the usual $<$ restricted to $[n]$, interpreting, for every $u \in U$, a unary relation symbol P_u as the set

$$[\![P_u]\!] = \{i \in [n] \mid u \in \alpha_i\}$$

of positions in s where u occurs. Forming unary predicate symbols P_u from elements u of a string symbol α is "unconventional" (Vu et al., 2018), the custom being instead to name unary predicates P_α after the string symbol α in its entirety (not generally assumed to be a set with noteworthy elements). This shift from α to an element $u \in \alpha$ is consequential, but preserves the Büchi-Elgot-Trakhtenbrot theorem characterizing regular languages as the sets of strings definable in *Monadic Second-Order Logic* over strings (e.g., Libkin, 2004, Theorem 7.21). For any subset X of U, the X-structure $M_X[\rho_X(s)]$ associated with the X-reduct $\rho_X(s)$ of s is the U-structure $M_U[s]$ with P_u restricted to $u \in X$.

Transition signatures add a bit more information about the set U of subscripts u on unary predicates

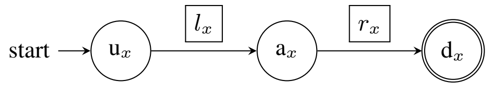

Figure 2: Interval $\boxed{u_x, l_x}\boxed{a_x, r_x}\boxed{d_x}$ as an automaton

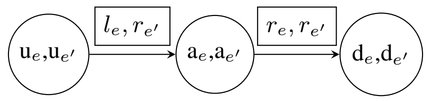

Figure 3: The shortest (middle) path in Figure 1

P_u, separating acts a from label-value pairs, and specifying the set $\mathsf{af}(a)$ of labels whose values an act a can *af*fect. Linked in section 2 to connections in nerve nets from input cells (acts) to inner cells (labels), the function af motivates the compression of X-reducts $\rho_X(s)$ of a string s, based on two dicta that bring up inertia

- *no time without change*

- *no change without force*

(meaning: *no* stuttering stative boxes *nor* empty boxes of acts). Compressing reducts deviates from the convention in institutions of using reducts for the contravariant functor **Mod**, altering a model's universe (of string positions) and damaging a property called amalgamation that is of some interest (e.g., Diaconescu, 2012; Sannella and Tarlecki, 2015). That damage is illustrated dramatically by the thirteen Allen interval relations from the conjunction of two Allen intervals; in pictures, Figure 1 from section 3 arises from Figure 2 with $x \in \{e, e'\}$ (e.g., Fernando, 2020). Without compression, Figure 1 would collapse to its shortest path, Figure 3, with e and e' marching in lockstep (born at the same time, and died at the same time).

Initial and final states are designated in Figure 2 to form a finite automaton, pointing more generally to the matter of computing constraints on strings beyond the reach of af. The clues from $\mathsf{af}(a)$ fall short of a specification of a's effects, never mind its preconditions. This is where the Σ-sentences φ from the functor Sen come in, each of which defines, via the relation $\models_\Sigma$, a set

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

of strings that we can assume is accepted by some finite automaton, provided we are careful enough with our choice of $\mathit{Sen}(\Sigma)$. The aforementioned Büchi-Elgot-Trakhtenbrot theorem provides an obvious candidate, but a number of representations of regular languages (beginning with Kleene (1956)'s regular expressions) are known. The pay-off from working with such representations is that the entailment from φ to ψ given by the inclusion

$$\mathbf{Mod}_\Sigma(\varphi) \subseteq \mathbf{Mod}_\Sigma(\psi)$$

of two regular languages is decidable. (Inclusion between say, context-free languages is not.)

There is no shortage of finite-state toolkits about. Mechanical support for interval reasoning in temporal annotation in TimeML (e.g., Pustejovsky et al., 2010) is described in Woods and Fernando (2018), based on a simplification of the string in Figure 2 to $\boxed{}\boxed{x}\boxed{}$, construable here as

$$\boxed{(x,0)}\boxed{(x,1)}\boxed{(x,0)}$$

with two values (0,1), rather than three (u,a,d). To represent acts such as buy(X,Y) along with their preconditions and effects, it is natural to box records and acts, connected by more interesting choices of af than those explored in section 3. But already with a simple interval x, its different representations raise the problem of semantic interoperability. We can formulate translations between representations two ways:

(i) within an institution, the **Sign**-morphisms in which may go beyond inclusions $\subseteq$ that **Mod** turns into X-projections, or

(ii) between institutions, each of which can be kept simple, if (as with signatures in **Sign**) there can be another to improve it.

The possibility in (ii) of multiple institution*s* points to logical pluralism (e.g., Kutz et al., 2010), cautioning against turning Definitions 1-3 into a single institution where all signatures can be found (and justifying some vagueness about what **Sign**, **Mod** and *Sen* precisely are). That said, any institution must meet a *Satisfaction Condition* asserting that for any **Sign**-morphism $\sigma : \Sigma \to \Sigma'$, Σ'-model s' and Σ-sentence φ,

$$s' \models_{\Sigma'} Sen(\sigma)(\varphi) \iff \mathbf{Mod}(\sigma')(s') \models_\Sigma \varphi.$$

For the special case of $\Sigma = (\mathbf{L}, \mathcal{V}, \mathbf{Act}, \mathsf{af})$ and $\Sigma' = (\mathbf{L}', \mathcal{V}', \mathbf{Act}', \mathsf{af}')$ where

$$\mathbf{L} \subseteq \mathbf{L}' \text{ and } \mathcal{V}' \restriction \mathbf{L} = \mathcal{V} \text{ and } \mathbf{Act} \subseteq \mathbf{Act}' \quad (21)$$

we can set $\mathbf{Mod}(\sigma)(s')$ to $\kappa_{voc(\Sigma)}(s')$ where the *vocabulary* $voc(\Sigma)$ *of* Σ is the set

$$voc(\Sigma) \ := \ \mathbf{Act} \cup \{(l,v) \mid l \in \mathbf{L} \text{ and } v \in \mathcal{V}(l)\}$$

of acts and label-value pairs, some subsets of which go into the set

$$\mathcal{B}_\Sigma \ := \{a \cup r \mid a \subseteq \mathbf{Act} \text{ and } r \in Q(\Sigma)\}$$

of Σ-*boxes* that are strung together into Σ-models $s \in \mathcal{B}_\Sigma{}^+$. Construing a Σ'-model s' as the $voc(\Sigma')$-structure $M_{voc(\Sigma')}[s']$ defined above, we can apply the translation scheme machinery in Makowsky (2004) to analyze $\kappa_{voc(\Sigma)}(s')$ as well as the Σ'-sentence $Sen(\sigma)(\varphi)$, abbreviated $\langle\sigma\rangle\varphi$, such that

$$s' \models_{\Sigma'} \langle\sigma\rangle(\varphi) \iff \kappa_{voc(\Sigma)}(s') \models_\Sigma \varphi.$$

The idea is $\kappa_{voc(\Sigma)}(s')$ restricts $M_{voc(\Sigma')}[s']$'s universe to string positions x satisfying the disjunction

$$\begin{aligned}\phi_\Sigma(x) \ \ := \ & \chi_{\mathbf{Act}}(x) \vee \chi'_{voc(\Sigma)\setminus\mathbf{Act}}(x) \vee \\ & \exists y(xSy \wedge \chi_{\mathbf{Act}}(y))\end{aligned}$$

where $\chi_{\mathbf{Act}}(x)$ says an act is done at x

$$\chi_{\mathbf{Act}}(x) \ := \bigvee_{a \in \mathbf{Act}} P_a(x)$$

while $\chi'_B(x)$ says some binding from B holds at x but not at x's successor

$$\chi'_B(x) \ := \bigvee_{u \in B} (P_u(x) \wedge \neg\exists y(xSy \wedge P_u(y))$$

(amounting to a B-discernible change at x), where S is the usual successor relation definable from $<$

$$xSy \ := \ x < y \wedge \neg\exists z(x < z \wedge z < y). \quad (22)$$

It is convenient here that $<$, rather than S, is primitive, as $\kappa_{voc(\Sigma)}(s')$ simply restricts $<$ to ϕ_Σ, and similarly with P_u, for $u \in voc(\Sigma)$. Not so with S, which the translation machinery analyzes as (22).

What about **Sign**-morphisms $\sigma : \Sigma \to \Sigma'$ for which the inclusions in (21) above do not hold? It suffices that σ come with a function $f_\sigma : \mathcal{B}_{\Sigma'} \to \mathcal{B}_\Sigma$ reducing a Σ'-box α' to a Σ-box $f_\sigma(\alpha')$, which we can extend homomorphically to $\mathcal{B}_{\Sigma'}{}^* \to \mathcal{B}_\Sigma{}^*$ before compressing by either $d^\square$ provided $voc(\Sigma) \subseteq \mathbf{Act}$, or $\kappa^{\mathbf{Act}}$ otherwise. The resulting composition is the $voc(\Sigma)$-projection $\kappa_{voc(\Sigma)}$ in case $f_\sigma(\alpha') = \alpha' \cap voc(\Sigma)$. In general, the point is to apply $\kappa_{voc(\Sigma)}$ after a map f_σ which adds *no* information in that for every Σ'-box α',

$$\alpha' \vdash f_\sigma(\alpha')$$

where $\vdash$ is a suitable notion of entailment. I hope to write elsewhere about some interesting examples of $\vdash$ (as well as f_σ), and what these have to do with Kleene (1956), in particular, with changes to the m-tuple $(s_1, \ldots, s_m)$ specifying the number of values that the inner cells $\mathcal{M}_1, \ldots, \mathcal{M}_m$ can take.

Bibiliographic note Connections between the present work and action signatures in M. Gelfond and V. Lifschitz 1998 (Action languages, Linköping Electronic Articles in Computer and Information Science, 3:16) are described in a companion paper, T. Fernando 2022 (Action signatures and finite-state variations, Proc ESSLLI Workshop: AREA II, Annotation, Recognition and Evaluation of Actions), where the relation (21) above between transition signatures Σ and Σ' is generalized to incorporate a notion of blurring (turning the records making up the stative part $Q(\Sigma)$ into record types).

References

J.F. Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

K.R. Beesley and L. Karttunen. 2003. *Finite State Morphology*, volume 2900 of LNCS. CSLI, Stanford.

G.N. Carlson. 1995. Truth conditions of generic sentences: Two contrasting views. In G.N. Carlson and F.J. Pelletier, editors, *The Generic Book*, pages 224–237. University of Chicago 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.

R. Diaconescu. 2012. Three decades of institution theory. In J-Y Béziau, editor, *Universal Logic: An Anthology*, pages 309–322. Springer.

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. 2020. Temporal representations with and without points. In R. Loukanova, editor, *Logic and Algorithms in Computational Linguistics 2018*, pages 45–66. Springer.

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

J. Heinz and J. Sempere, editors. 2016. *Topics in Grammatical Inference*. Springer-Verlag.

C. de la Higuera. 2010. *Grammatical Inference: Learning Automata and Grammars*. Cambridge University Press.

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. 2013. The time of my life. Available at https://lucian.uchicago.edu/blogs/elucidations/files/2013/08/KampTheTimeOfMyLife.pdf.

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

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.

O. Kutz, T. Mossakowski, and D. Lücke. 2010. Carnap, Goguen, and the Hyperontologies: Logical pluralism and heterogeneous structuring in ontology design. *Logica Universalis*, 4:255–333.

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

J.A. Makowsky. 2004. Algorithmic uses of the Feferman-Vaught theorem. *Annals of Pure and Applied Logic*, 126:159–213.

W. S. McCulloch and W. H. Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.*, 5:115–133.

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

J. Pustejovsky, K. Lee, H. Bunt, and L. Romary. 2010. ISO-TimeML: An international standard for semantic annotation. In *Proc 7th International Conference on Language Resources and Evaluation (LREC'10)*, pages 394–397.

M.O. Rabin and D.S. Scott. 1959. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3:114–125.

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.

M.H. Vu, A. Zehfroosh, K. Strother-Garcia, M. Sebok, J. Heinz, and H.G. Tanner. 2018. Statistical relational learning with unconventional string models. *Frontiers in Robotics & AI*, 5.

D. Woods and T. Fernando. 2018. Improving string processing for temporal relations. In *Proc 14th Joint ISO-ACL Workshop on Interoperable Semantic Annotation (ISA-14)*, pages 76–86.