

Exploring Semantic Incrementality with Dynamic Syntax and Vector Space Semantics

Mehrnoosh Sadrzadeh,¹ Matthew Purver,¹ Julian Hough,¹ Ruth Kempson²

¹School of Electronic Engineering and Computer Science,
Queen Mary University of London

²Department of Philosophy, King's College London
mehrnoosh.sadrzadeh@qmul.ac.uk

Abstract

One of the fundamental requirements for models of semantic processing in dialogue is *incrementality*: a model must reflect how people interpret and generate language incrementally, and handle phenomena such as fragments, incomplete and jointly-produced utterances. We show that the incremental word-by-word parsing process of Dynamic Syntax (DS) can be assigned a compositional distributional semantics, with the composition operator of DS corresponding to the general operation of tensor contraction from multilinear algebra. We provide abstract semantic decorations for the nodes of DS trees, in terms of vectors, tensors, and sums thereof; using the latter to model the underspecified elements crucial to assigning partial representations during incremental processing. As a working example, we give an instantiation of this theory using plausibility tensors of compositional distributional semantics, and show how our framework can incrementally assign a semantic plausibility measure as it parses phrases and sentences.

1 Introduction

An incremental, word-by-word view on language processing is motivated by much empirical evidence from human-human dialogue. This evidence includes split, interrupted, and corrective utterances, see e.g. (Howes et al., 2011):

(1) A: Ray destroyed . . .

B: . . . the fuchsia. He never liked it. The roses he spared . . .

A: . . . this time.

In (1), the utterances are either inherently incomplete or potentially complete, with more than one agent contributing to the unfolding of a sequence, with in principle arbitrary speaker switch points and indefinite extendibility. In such cases, speakers and hearers must be processing the structural and semantic information encoded in each utterance incrementally. A second motivation comes from computational dialogue systems, where the ability to process incrementally helps speed up systems and provide more natural interaction (Aist et al., 2007). A third motivation comes from psycholinguistic results, even in individual language processing, which show that hearers can incrementally disambiguate word senses and resolve references, before sentences are complete and even using partial words and disfluent material to do so (Brennan and Schober, 2001). In (2a,b), the ambiguous word *dribbled* can be resolved to a particular sense early on, given the (*footballer* or *baby*) subject, without waiting for the rest of the sentence. A fourth comes from cognitive neuroscience and models such as Predictive Processing (Friston and Frith, 2015; Clark, 2015) which focus on agents' incremental ability to generate expectations and judge the degree to which they are met by observed input.

(2) a. The footballer dribbled the ball across the pitch.

b. The baby dribbled the milk all over the floor.

This work is licenced under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>

We use the framework of Dynamic Syntax (DS) for incremental grammatical and semantic analysis (Kempson et al., 2001; Cann et al., 2005; Kempson et al., 2016). DS has sufficient expressivity to capture the dialogue phenomena in (1) and has been used to provide incremental interpretation and generation for dialogue systems (Purver et al., 2011; Eshghi et al., 2017). Yet incremental disambiguation is currently beyond its expressive power; and while its framework is broadly predictive, it does not yet provide an explanation for how specific expectations can be generated or their similarity to observations measured.

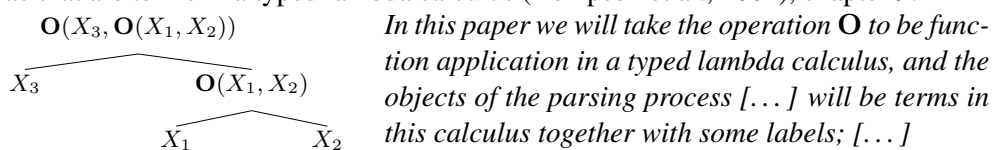
DS does not fix a special form of syntax and instead defines grammaticality directly in terms of incremental semantic tree growth. Symbolic methods are employed for labelling the contents of these trees, via terms either from an epsilon calculus (Kempson et al., 2001) or a suitable type theory with records (Purver et al., 2010). These symbolic approaches are not able to reflect the non-deterministic content of natural language forms, nor the way any initially unfixable interpretation, polysemy being rampant, can be narrowed down during the utterance interpretation process. For the same reason, the assigned term specifications do not provide a basis for the graded judgements that humans are able to make during processing to assess similarity to (or divergence from) expectations (Clark, 2015), to incrementally narrow down a word’s interpretation, or disambiguate its sense in the emerging context.

Non-determinisms of meaning and gradient similarity judgements are the stronghold of the so-called distributional or vector space semantics (Schütze, 1998; Lin, 1998; Curran, 2004; Salton et al., 1975). By modelling word meanings as vectors within a continuous space, such approaches directly express graded similarity of meaning (e.g. as distance or angle between vectors) and changes in interpretation (via movements of vectors within a space). Vector space semantics has been extended from words to phrases and sentences using different grammatical formalisms, e.g. Lambek pregroups, Lambek Calculus, and the CCG (Maillard et al., 2014; Krishnamurthy and Mitchell, 2013; Coecke et al., 2010; Coecke et al., 2013). It has, however, not been extended to incremental and dialogue formalisms such as DS.

In this paper, we address the above mentioned problems, by defining an incremental vector space semantic model for DS that can express non-determinism and similarity in word meaning, and yet keep incremental compositionality over conversational exchanges. As a working example, we instantiate this model using the plausibility instance of (Clark, 2013b) developed for a type-driven compositional distributional semantics, and show how it can incrementally assign a semantic plausibility measure as it performs word-by-word parses of phrases and sentences. We discuss how this ability enables us to incrementally disambiguate words using their immediate contexts and to model the plausibility of continuations and thus a hearer’s expectations.

2 Dynamic Syntax and its Semantics

In its original form, Dynamic Syntax (DS) provides a strictly incremental formalism relating word sequences to semantic representations. Conventionally, these are seen as trees decorated with semantic formulae that are terms in a typed lambda calculus (Kempson et al., 2001), chapter 9:



This allows us to give analyses of the semantic output of the word-by-word parsing process in terms of partial semantic trees, in which nodes are labelled with type Ty and semantic formula Fo , or with requirements for future development (e.g. $?Ty$, $?Fo$), and with a pointer \diamond indicating the node currently under development. This is shown in Figure 1 for the simple sentence *Mary likes John*. Phenomena such as conjunction, apposition and relative clauses are analysed via LINKed trees (corresponding to semantic conjunction). **For reasons of space we do not present an original DS tree here; an example of a non-restrictive relative clause linked tree labelled with vectors is presented in Figure 3.**

The DS formalism is in fact considerably more general. To continue the quotation above:

[...] it is important to keep in mind that the choice of the actual representation language is not central to the parsing model developed here. [...] For instance, we may take X_1, X_2, X_3 to be feature structures and the operation O to be unification, or X_1, X_2, X_3 to be lambda terms

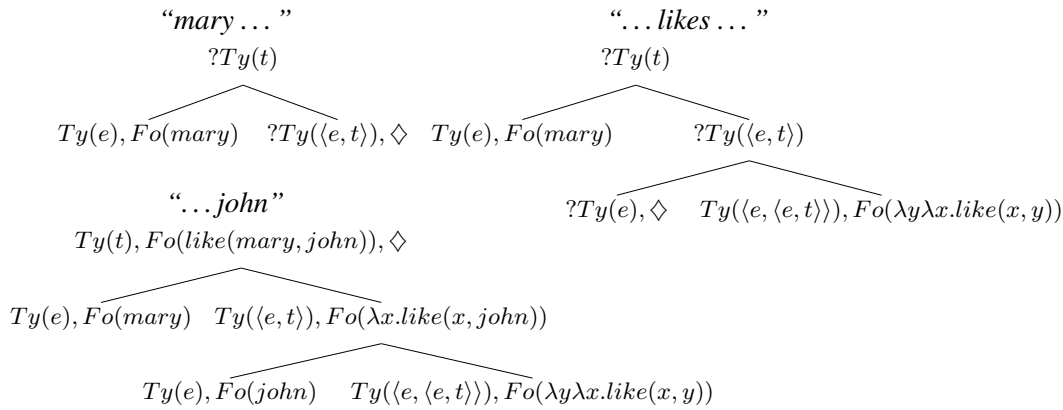


Figure 1: DS parsing as semantic tree development, for the simple sentence “*mary likes john*”.

and **O** Application, or X_1, X_2, X_3 to be labelled categorial expressions and **O** Application: Modus Ponens, or X_1, X_2, X_3 to be DRSs and **O** Merging.

Indeed, in some variants this generality is exploited; for example, Purver et al. (2010) outline a version in which the formulae are *record types* in Type Theory with Records (TTR) (Cooper, 2005); and Hough and Purver (2012) show how this can confer an extra advantage – the incremental decoration of the *root* node, even for partial trees, with a maximally specific formula via type inference, using the TTR merge operation \wedge as the composition function. In the latter account, underspecified record types decorate requirement nodes, containing a type judgement with the relevant type (e.g. $[x : e]$ at type $?Ty(e)$ nodes). Hough and Purver (2017) show that this underspecification can be given a precise semantics through record type lattices: the dual operation of merge, the minimum common super type (or join) \vee is required to define a (probabilistic) distributive record type lattice bound by \wedge and \vee . The interpretation process, including reference resolution, then takes the incrementally built top-level formula and checks it against a type system (corresponding to a world model) defined by a record type lattice. Implicitly, the record type on each node in a DS-TTR tree can be seen to correspond to a potential set of type judgements as sub-lattices of this lattice, with the appropriate underspecified record type (e.g. $[x : e]$) as their top element, with a probability value for each element in the probabilistic TTR version. In this paper, we show how equivalent underspecification, and narrowing down of meaning over time — but with the additional advantages inherent in vector space models, e.g. similarity judgements — can be defined for vector space representations with analogous operations to \wedge and \vee .

3 Compositional Vector Space Semantics for DS

Vector space semantics are commonly instantiated via lexical co-occurrence, based on the *distributional hypothesis* that meanings of words are represented by the distributions of the words around them (and often described by a quote from Firth, that “you shall know a word by the company it keeps” (Firth, 1957)). This can be implemented by creating a co-occurrence matrix (Rubenstein and Goodenough, 1965), whose columns are labelled by context words and whose rows by target words; the entry of the matrix at the intersection of a context word c and a target word t is a function (such as TF-IDF or PPMI) of the number of times t occurred in the context of c (as defined via e.g. a lexical neighbourhood window, a dependency relation, etc.). The meaning of each target word is represented by its corresponding row of the matrix. These rows are embedded in a vector space, where the distances between the vectors represent degrees of semantic similarity between words (Curran, 2004; Lin, 1998; Schütze, 1998).

Distributional semantics has been extended from word level to sentence level, where a compositional operation acts on the vectors of the words to produce a vector for the sentence. Existing models vary from using simple additive and multiplicative compositional operations (Mitchell and Lapata, 2010) to compositional operators based on fully fledged categorial grammar derivations, e.g. pregroup grammars (Coecke et al., 2010; Clark, 2013b) or CCG (Baroni et al., 2014; Maillard et al., 2014; Krishnamurthy

and Mitchell, 2013). However, the work done so far has not been directly compatible with incremental processing: this paper is the first attempt to develop such an incremental semantics, using a framework not based on a categorial grammar, i.e. one in which a full categorial analysis of the phrase/sentence is not the obligatory starting point.

Compositional vector space semantic models have a complementary property to DS. Whereas DS is agnostic to its choice of semantics, compositional vector space models are agnostic to the choice of the syntactic system. Coecke et al. (2010) show how they provide semantics for sentences based on the grammatical structures given by Lambek’s pregroup grammars (Lambek, 1997); Coecke et al. (2013) show how this semantics also works starting from the parse trees of Lambek’s Syntactic Calculus (Lambek, 1958); Wijnholds (2017) shows how the same semantics can be extended to the Lambek-Grishin Calculus; and (Maillard et al., 2014; Krishnamurthy and Mitchell, 2013; Baroni et al., 2014) show how it works for Combinatory Categorial Grammar trees. These semantic models homomorphically map the concatenation and slashes of categorial grammars to tensors and their evaluation/application/composition operations to tensor contraction.

In DS terms, structures X_1, X_2, X_3 are mapped to general higher order tensors, e.g. as follows:

$$\begin{aligned} X_1 &\mapsto T_{i_1 i_2 \dots i_n} && \in V_1 \otimes V_2 \otimes \dots \otimes V_n \\ X_2 &\mapsto T_{i_n i_{n+1} \dots i_{n+k}} && \in V_n \otimes V_{n+1} \otimes \dots \otimes V_{n+k} \\ X_3 &\mapsto T_{i_{n+k} i_{n+k+1} \dots i_{n+k+m}} && \in V_{n+k} \otimes V_{n+k+1} \otimes \dots \otimes V_{n+k+m} \end{aligned}$$

Each $T_{i_1 i_2 \dots i_n}$ abbreviates the linear expansion of a tensor, which is normally written as follows:

$$T_{i_1 i_2 \dots i_n} \equiv \sum_{i_1 i_2 \dots i_n} C_{i_1 i_2 \dots i_n} e_1 \otimes e_2 \otimes \dots \otimes e_n$$

for e_i a basis of V_i and $C_{i_1 i_2 \dots i_n}$ its corresponding scalar value. The \mathbf{O} operations are mapped to contractions between these tensors, formed as follows:

$$\begin{aligned} \mathbf{O}(X_1, X_2) &\mapsto T_{i_1 i_2 \dots i_n} T_{i_n i_{n+1} \dots i_{n+k}} \\ &\in V_1 \otimes V_2 \otimes \dots \otimes V_{n-1} \otimes V_{n+1} \otimes \dots \otimes V_{n+k} \\ \mathbf{O}(X_3, \mathbf{O}(X_1, X_2)) &\mapsto T_{i_1 i_2 \dots i_n} T_{i_n i_{n+1} \dots i_{n+k}} T_{i_{n+k} i_{n+k+1} \dots i_{n+k+m}} \\ &\in V_1 \otimes V_2 \otimes \dots \otimes V_{n-1} \otimes V_{n+1} \otimes \dots \otimes V_{n+k-1} \otimes V_{n+k+1} \otimes \dots \otimes V_{n+k+m} \end{aligned}$$

In their most general form presented above, these formulae are large and the index notation becomes difficult to read. In special cases, however, it is often enough to work with spaces of rank around 3. For instance, the application of a transitive verb to its object is mapped to the following contraction:

$$T_{i_1 i_2 i_3} T_{i_3} = \left(\sum_{i_1 i_2 i_3} C_{i_1 i_2 i_3} e_1 \otimes e_2 \otimes e_3 \right) \left(\sum_{i_3} C_{i_3} e_3 \right) = \sum_{i_1 i_2} C_{i_1 i_2 i_3} C_{i_3} e_1 \otimes e_2$$

This is the contraction between a cube $T_{i_1 i_2 i_3}$ in $X_1 \otimes X_2 \otimes X_3$ and a vector T_{i_3} in X_3 , resulting in a matrix in $T_{i_1 i_2}$ in $X_1 \otimes X_2$.

We take the DS propositional type $Ty(t)$ to correspond to a sentence space S , and the entity type $Ty(e)$ a word space W . Given vectors T_i^{mary}, T_k^{john} in W and the (cube) tensor T_{ijk}^{like} in $W \otimes S \otimes W$, the tensor semantic trees of the DS parsing process of *Mary likes John* become as in Fig. 2.¹

A very similar procedure is applicable to the linked structures, where conjunction can be interpreted by the μ map of a Frobenius algebra over a vector space, e.g. as in (Kartsaklis, 2015), or as composition of the interpretations of its two conjuncts, as in (Muskens and Sadrzadeh, 2016). The μ map has also been used to model relative clauses (Clark et al., 2013; Sadrzadeh et al., 2013; Sadrzadeh et al., 2014). It *combines* the information of the two vector spaces into one. Figure 2 shows how it combines the information of two contracted tensors $T_i^{mary} T_{ij}^{sleep}$ and $T_i^{mary} T_{ij}^{snore}$.

¹There has been much discussion about whether sentence and word spaces should be the same or separate. In previous work, we have worked with both cases, i.e. when $W \neq S$ and when $W = S$.

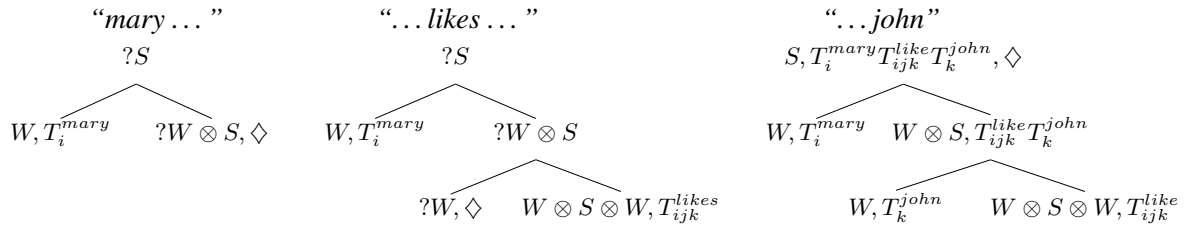


Figure 2: A DS with Vector Space Semantics parse of ‘Mary likes John’.

DS *requirements* can now be treated as requirements for tensors of a particular order (e.g. $?W$, $?W \otimes S$ as above). If we can give these suitable vector-space representations, we can then provide an analogue to Hough and Purver (2012)’s incremental type inference procedure, allowing us to compile a partial tree to specify its overall semantic representation (at its root node). One alternative would be to interpret them as picking out an element which is *neutral* with regards to composition: the unit vector/tensor of the space they annotate. A more informative alternative would be to interpret them as enumerating all the possibilities for further development. This can be derived from all the word vector and phrase tensors of the space under question — i.e. all the word and phrases whose vectors and tensors live in W and in $W \otimes S$ in this case — by taking either the *sum* T^+ or the *direct sum* T^\oplus of these vectors/tensors. Summing will give us one vector/tensor, accumulating the information encoded in the vectors/tensors of each word/phrase; direct summing will give us a tuple, keeping this information separate from each other. This gives us the equivalent of a sub-lattice of the record type lattices described in (Hough and Purver, 2017), with the appropriate underspecified record type as the top element, and the attendant advantages for incremental probabilistic interpretation.

These alternatives all provide the desired compositionality, but differ in the semantic information they

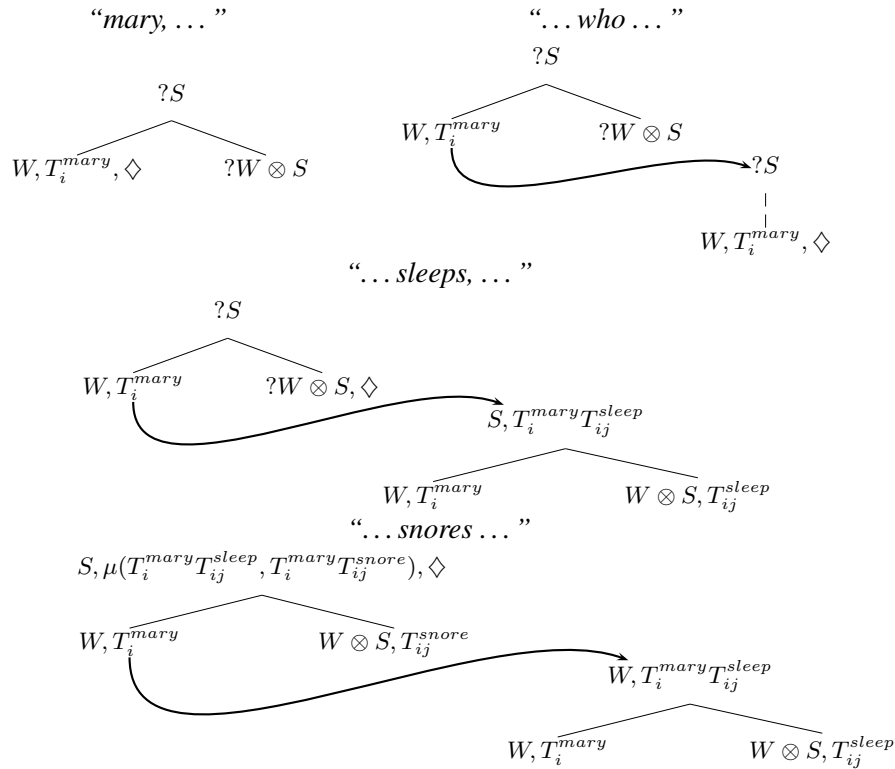


Figure 3: A DS with Vector Space Semantics parse of ‘Mary, who sleeps, snores’.

contribute. The use of the identity provides no semantic information; the sum gives information about the “average” vector/tensor expected on the basis of what is known about the language and its use in context (encoded in the vector space model); the direct sum enumerates the possibilities. In each case, more semantic information can then arrive later as more words are parsed. The best alternative will depend on task and implementation: in the next section, we give a working example using the sum operation.

4 Incremental Plausibility: a working example

In order to exemplify the abstract tensors and tensor contraction operations of the model and provide a proof of concept for its applicability to semantic incrementality, we characterise the incremental disambiguation of the *The footballer dribbled....* example. This example is worked out in the instance of the compositional distributional semantics introduced in (Clark, 2013b) and implemented in (Polajnar et al., 2014), intended to model *plausibility*. In this instance, S is a two dimensional space with basis vectors true \top and false \perp . Sentences that are highly plausible have a vector representation close to the \top basis; highly implausible sentences have one close to the \perp basis. As an illustrative example, we take W to be the following 4×4 matrix based on co-occurrence counts:²

	infant	nappy	pitch	goal
baby	34	10	0	0
milk	10	1	0	0
footballer	0	0	11	52
ball	0	1	27	49

As an example, consider the row corresponding to *baby*: this gives us a vector with the linear expansion $\sum_i C_i^{baby} e_i$, for $e_i \in \{\textit{infant}, \textit{nappy}, \textit{pitch}, \textit{goal}\}$ a basis vector of W and C_i^{baby} its corresponding scalar value. The value $C_1^{baby} = 3$ represents the number of times *baby* occurred in the same piece of text as *nappy*; the value $C_4^{baby} = 0$ represents the number of times *baby* occurred in the same excerpt as *goal*, e.g. as the subjects of *wore nappy* or *crawled into a goal*.

Intransitive verbs v will have matrix representations with linear expansion $\sum_{ij} C_{ij}^v e_i \otimes e_j$ with e_i a basis vector of W and e_j a basis vector of S . A high value for v on the basis $\langle e_i, \top \rangle$ means that it is highly plausible that v has the property e_i ; a high value at the $\langle e_i, \perp \rangle$ means that it is highly implausible that v has property e_i . For example, consider the verbs *vomit*, *score*, *dribble* in their intransitive roles: T^{score} has a high value at $\langle \textit{goal}, \top \rangle$, since it is highly plausible that things that are scored are goals; and a high entry at $\langle \textit{nappy}, \perp \rangle$, since it is highly implausible that things that wear nappies (e.g. babies) score. T^{vomit} has an opposite plausibility distribution for infant and nappy wearing agents. $T^{dribble}$ is a mixture of these two, since both nappy wearing and goal scoring agents do it, but in different senses. Here, we instantiate the matrix purely from text co-occurrence, approximating plausibility from co-occurrence of verb and entity in the same text excerpt and implausibility from lack thereof, i.e. occurrence of verb without the entity. Other methods could of course be used, e.g. using dependency parse information to show verb-agent relations directly; or learning entries via regression (Polajnar et al., 2014). Note that while this makes our plausibility and implausibility degrees dependent, and the two dimensional S can therefore be reduced to a one dimensional one, the theory supports spaces of any dimension, so we present values and computations for both dimensions to illustrate this.

	$\langle \textit{infant}, \top \rangle$	$\langle \textit{infant}, \perp \rangle$	$\langle \textit{nappy}, \top \rangle$	$\langle \textit{nappy}, \perp \rangle$	$\langle \textit{pitch}, \top \rangle$	$\langle \textit{pitch}, \perp \rangle$	$\langle \textit{goal}, \top \rangle$	$\langle \textit{goal}, \perp \rangle$
vomit	10	2	9	3	3	9	0	12
score	1	7	0	8	7	1	8	0
dribble	14	6	15	5	18	2	19	1

The interpretation of an intransitive sentence, such as *Babies vomit* is calculated as follows:

$$\begin{aligned}
 T^{\textit{babies vomit}} &= T_i^{\textit{babies}} T_{ij}^{\textit{vomit}} = (C_1^{baby} C_{11}^{\textit{vomit}} + C_2^{baby} C_{21}^{\textit{vomit}} + C_3^{baby} C_{31}^{\textit{vomit}} + C_4^{baby} C_{41}^{\textit{vomit}}) \top + \\
 &\quad (C_1^{baby} C_{12}^{\textit{vomit}} + C_2^{baby} C_{22}^{\textit{vomit}} + C_3^{baby} C_{32}^{\textit{vomit}} + C_4^{baby} C_{42}^{\textit{vomit}}) \perp \\
 &= (34 \times 10 + 10 \times 9) \top + (34 \times 2 + 10 \times 3) \perp \\
 &= 430 \top + 98 \perp
 \end{aligned}$$

²For illustrative purposes, the co-occurrence counts are taken from random excerpts of up to 100 sentences, taken from the BNC; a full implementation would of course use larger datasets.

Similar calculations provide us with the following sentence representations:

$$\begin{aligned}
T^{\text{babies score}} &= 34\top + 318\perp \\
T^{\text{babies dribble}} &= 626\top + 254\perp \\
T^{\text{footballers vomit}} &= 33\top + 723\perp \\
T^{\text{footballers score}} &= 493\top + 11\perp \\
T^{\text{footballers dribble}} &= 483\top + 74\perp
\end{aligned}$$

It follows that *Babies vomit* is more plausible than *Footballers vomit*, *Footballers score* is more plausible than *Babies score*, but *Babies dribble* and *Footballers dribble* have more or less the same degree of plausibility.

A transitive verb such as *control* will have a tensor representation as follows:

$$T^{\text{control}} = \sum_{ijk} C_{ijk}^{\text{control}} e_i \otimes e_j \otimes e_k$$

for e_i, e_k basis of W and e_j either \top or \perp . Suppose that *control* has a 1 entry value at *pitch* and *goal* with $e_j = \top$ and a low or zero entry everywhere else. It is easy to show that the sentence representation of *Footballers control balls* is much more plausible than that of *Babies control balls*.

$$\begin{aligned}
T^{\text{footballers control balls}} &= T_i^{\text{footballers}} T_{ijk}^{\text{control}} T_k^{\text{balls}} \\
&= C_i^{\text{footballer}} C_k^{\text{ball}} \langle \text{pitch} \rangle C_{ijk}^{\text{control}} (\langle \text{pitch}, \top, \text{pitch} \rangle + \langle \text{pitch}, \top, \text{goal} \rangle) \\
&\quad + C_i^{\text{footballer}} C_k^{\text{ball}} \langle \text{goal} \rangle C_{ijk}^{\text{control}} (\langle \text{goal}, \top, \text{pitch} \rangle + \langle \text{goal}, \top, \text{goal} \rangle) \\
&= 6866\top \\
T^{\text{babies control balls}} &= T_i^{\text{babies}} T_{ijk}^{\text{control}} T_k^{\text{balls}} = 0
\end{aligned}$$

In an unfinished utterance, such as *babies . . .*, parsing will first derive a semantic tree containing the vector for *babies* and a tensor for $?W \otimes S$; then we tensor contract the two to obtain a vector in $?S$. The underspecified tensor in $W \otimes S$ is computed by summing all known elements of $W \otimes S$:

$$T_{ij}^+ = T^{\text{vomit}} + T^{\text{score}} + T^{\text{dribble}} + T^{\text{control baby}} + T^{\text{control milk}} + T^{\text{control footballer}} + T^{\text{control ball}}$$

The tensor contraction of this with the vector of *babies* provides us with the meaning of the utterance:

$$T_i^{\text{babies}} T_{ij}^+$$

Similar calculations to the previous cases show that plausibility increases when moving from the incomplete utterance $T_i^{\text{babies}} T_{ij}^+$ to the complete one $T_i^{\text{babies}} T_{ij}^{\text{vomit}}$. Conceptually speaking, the incomplete phrase will be a dense, high-entropy vector with nearly equal values on \top and \perp , whereas the complete phrase (or the more complete phrase), will result in a sparser vector with more differential values on \top and \perp . Continuation with a less plausible verb, e.g. *score* would result in a reduction in plausibility; and different transitive verb phrases would of course have corresponding different effects. We therefore cautiously view this as an initial step towards a model which can provide the “error signal” feedback assumed in models of expectation during language interpretation (Clark, 2015).

5 Nondeterminism of Meaning and Incremental Disambiguation

5.1 Incremental Disambiguation

Distributional semantics comes with a straightforward algorithm for acquisition of word meaning, but when a word is ambiguous its vector representation becomes a mixture of the representations of its different senses. Post processing of these vectors is needed to obtain different representations for each sense (Schütze, 1998; Kartsaklis and Sadrzadeh, 2013). Given vectors for individual senses, our setting can incrementally disambiguate word meanings as the sentence is processed. For instance, we can

incrementally determine that in “footballers dribble”, “dribble” means “control the ball”; while in “babies dribble” it means “drip”. This is done by computing that “babies dribble_{drip}” is more plausible than “babies dribble_{control}”, and also that “footballers dribble_{control}” is more plausible than “footballers dribble_{drip}”. Note that this disambiguation can be made before the sentence is complete: in “her fingers tapped on her i-pad”, or “the police tapped his phone”, the combination of subject and verb alone can (given suitable vectors and tensors) give information about the relative plausibility of the readings of “tapped” as “knocked” or “intercepted”. This can then be strengthened when the object is parsed (or, indeed, weakened or even reversed, depending on the object).

The above examples are taken from the disambiguation dataset of (Kartsaklis et al., 2013). Parts of this dataset has been tested on the plausibility model of (Clark, 2013b) by (Polajnar et al., 2014), where it has been shown that plausibility implementations of verb tensors do a better job in disambiguating them. Repeating this task in our model to experimentally validate the incremental disambiguation hypothesis constitutes work in progress.

5.2 Incremental Expectation

Using our model on examples such as the above, we can also incrementally compute plausibility of possible continuations. Consider again the “dribble” example: after parsing *Footballers dribble*, we can calculate not merely that the verb’s interpretation can be narrowed down in the presence of the subject, but also that the continuation *ball* would be very plausible, and the continuation *milk* very implausible. A similar computation provides us with the plausible continuations for *Police intercept vs Fingers knock*. If we are using the (direct) sum method to assign overall plausibility to the unfinished sentence, the plausibility values of the possible continuations have already been calculated; here we need only inspect the particular values of interest. Using this method, we can therefore explain how people assign shifting expectations as parsing proceeds, and make interim probabilistic evaluations on the basis thereof – giving us a basis for a model embodying the ‘predictive processing’ stance of (Clark, 2013a). Again, experimentally evaluating this hypothesis is left to future work.

6 Discussion

This model gives us a basis for incremental interpretation via compositional, grammar-driven vector space semantics. The particular instantiation outlined above assigns sentence representations in only a two-dimensional plausibility space, but the framework generalises to any vector space. Our intention is to extend this to more informative spaces, and integrate with the incremental probabilistic approaches to interpretation (e.g. Hough and Purver (2017)’s approach to reference resolution).

One important step will be to adapt the model for incremental *generation*. In the original formulation of DS generation (Purver and Kempson, 2004), generation is defined as a process of DS parsing, along with a check against a *goal tree*. At each generation step, every word in the vocabulary is tested to check if it is parseable from the current parse state; those which can be parsed are tested, with the resulting DS tree being checked to see if it subsumes the goal tree. If it does subsume it, then the parsed word can be generated as output; when the current tree and goal tree match, generation is complete and the process halts. (Hough and Purver, 2012) updated this to use a goal *concept* as a TTR record type, with the subsumption check now testing whether a DS-TTR tree’s top-level record type is a proper supertype of (i.e. subsumes) the current goal record type. Given the equivalence of our proposed model to (Hough and Purver, 2012)’s parsing process described above, the only additional apparatus required for generation for DS with Vector Space Semantics is the use of a goal *tensor*, and a characterisation of subsumption between two tensors. For the latter, we intend to look into a distributional characterisation of inclusion (Kartsaklis and Sadrzadeh, 2016), in the spirit of a real-valued measure of relevance proposed in probabilistic type theory by (Hough and Purver, 2017). Other approaches to this are exploring type theory and vector space semantics hybrids such as (Asher et al., 2017).

References

- G. Aist, J. Allen, E. Campana, C.A. Gomez Gallo, S. Stoness, M. Swift, and M.K. Tanenhaus. 2007. Incremental dialogue system faster than and preferred to its nonincremental counterpart. In *Proceedings of the 29th Annual Conference of the Cognitive Science Society*.
- Nicholas Asher, Marta Abrusan, and Tim Van de Cruys. 2017. Types, meanings and co-composition in lexical semantics. In *Modern Perspectives in Type-Theoretical Semantics*, pages 135–161. Springer.
- M. Baroni, R. Bernardi, and R. Zamparelli. 2014. Frege in space: A program of compositional distributional semantics. *Linguistic Issues in Language Technology*, 9.
- S.E. Brennan and M.F. Schober. 2001. How listeners compensate for disfluencies in spontaneous speech. *Journal of Memory and Language*, 44(2):274–296.
- Ronnie Cann, Ruth Kempson, and Lutz Marten. 2005. *The Dynamics of Language: An Introduction. Syntax and Semantics. Volume 35*. ERIC.
- Stephen Clark, Bob Coecke, and Mehrnoosh Sadrzadeh. 2013. The frobenius anatomy of relative pronouns. In *13th Meeting on Mathematics of Language (MoL)*, pages 41–51, Stroudsburg, PA. Association for Computational Linguistics.
- Andy Clark. 2013a. Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences*, 36(3):181–204.
- Stephen Clark. 2013b. Vector space models of lexical meaning. In Mehrnoosh Sadrzadeh Chris Heunen and Edward Grefenstette, editors, *Quantum Physics and Linguistics: A Compositional, Diagrammatic Discourse*, pages 359–377. Oxford University Press, 1st edition.
- Andy Clark. 2015. *Surfing uncertainty: Prediction, action, and the embodied mind*. Oxford University Press.
- B. Coecke, M. Sadrzadeh, and S. Clark. 2010. Mathematical foundations for a compositional distributional model of meaning. *Linguistic Analysis*, 36:345–384.
- Bob Coecke, Edward Grefenstette, and Mehrnoosh Sadrzadeh. 2013. Lambek vs Lambek: Functorial vector space semantics and string diagrams for Lambek calculus. *Ann. Pure and Applied Logic*, 164(11):1079–1100.
- Robin Cooper. 2005. Records and record types in semantic theory. *J. Logic and Computation*, 15(2):99–112.
- J. Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, School of Informatics, University of Edinburgh.
- Arash Eshghi, Igor Shalymov, and Oliver Lemon. 2017. Bootstrapping incremental dialogue systems from minimal data: the generalisation power of dialogue grammars. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2220–2230, Copenhagen, Denmark, September. Association for Computational Linguistics.
- J.R. Firth. 1957. A synopsis of linguistic theory 1930–1955. In *Studies in Linguistic Analysis*.
- Karl Friston and Christopher Frith. 2015. A duet for one. *Consciousness and cognition*, 36:390–405.
- Julian Hough and Matthew Purver. 2012. Processing self-repairs in an incremental type-theoretic dialogue system. In *Proc. 16th SemDial Workshop*, pages 136–144, Paris, France, September.
- Julian Hough and Matthew Purver, 2017. *Probabilistic Record Type Lattices for Incremental Reference Processing*, pages 189–222. Springer International Publishing.
- Christine Howes, Matthew Purver, Patrick G. T. Healey, Gregory J. Mills, and Eleni Gregoromichelaki. 2011. On incrementality in dialogue: Evidence from compound contributions. *Dialogue and Discourse*, 2(1):279–311.
- Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2013. Prior disambiguation of word tensors for constructing sentence vectors. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1590–1601.
- Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2016. Distributional inclusion hypothesis for tensor-based composition. *arXiv preprint arXiv:1610.04416*.

- Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. 2013. Separating disambiguation from composition in distributional semantics. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013*, pages 114–123.
- Dimitrios Kartsaklis. 2015. *Compositional Distributional Semantics with Compact Closed Categories and Frobenius Algebras*. Ph.D. thesis, Department of Computer Science, University of Oxford.
- Ruth Kempson, Wilfried Meyer-Viol, and Dov Gabbay. 2001. *Dynamic Syntax: The Flow of Language Understanding*. Blackwell, Oxford.
- Ruth Kempson, Ronnie Cann, Eleni Gregoromichelaki, and Stergios Chatzikyriakidis. 2016. Language as mechanisms for interaction. *Theoretical linguistics*, 42(3-4):203–276.
- Jayant Krishnamurthy and Tom M. Mitchell. 2013. Vector space semantic parsing: A framework for compositional vector space models. In *Proc. ACL Workshop on Continuous VSMs and their Compositionality*.
- J. Lambek. 1958. The mathematics of sentence structure. *American Mathematics Monthly*, 65:154–170.
- J. Lambek. 1997. Type grammars revisited. In *Proc. LACL 97*. Springer.
- D. Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics.
- J. Maillard, S. Clark, and E. Grefenstette. 2014. A type-driven tensor-based semantics for CCG. In *Proceedings of the Type Theory and Natural Language Semantics Workshop, EACL 2014*.
- Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34:1388–1439.
- Reinhard Muskens and Mehrnoosh Sadrzadeh. 2016. Context update for lamdas and vectors. In *LNCS Proceedings of the 9th International Conference on Logical Aspects of Computational Linguistics*, Nancy, December. Springer. to appear.
- Tamara Polajnar, Laura Rimell, and Stephen Clark. 2014. Using sentence plausibility to learn the semantics of transitive verbs. *CoRR*, abs/1411.7942.
- Matthew Purver and Ruth Kempson. 2004. Context-based incremental generation for dialogue. In *Natural language generation*, pages 151–160. Springer.
- Matthew Purver, Eleni Gregoromichelaki, Wilfried Meyer-Viol, and Ronnie Cann. 2010. Splitting the ‘I’s and crossing the ‘You’s: Context, speech acts and grammar. In *Proc. 14th SemDial Workshop*, pages 43–50, June.
- Matthew Purver, Arash Eshghi, and Julian Hough. 2011. Incremental semantic construction in a dialogue system. In *Proceedings of the Ninth International Conference on Computational Semantics, IWCS ’11*, pages 365–369, Stroudsburg, PA, USA. Association for Computational Linguistics.
- H. Rubenstein and J.B. Goodenough. 1965. Contextual Correlates of Synonymy. *Communications of the ACM*, 8(10):627–633.
- Mehrnoosh Sadrzadeh, Stephen Clark, and Bob Coecke. 2013. Frobenius anatomy of word meanings i: subject and object relative pronouns. *Journal of Logic and Computation*, 23:1293–1317.
- Mehrnoosh Sadrzadeh, Stephen Clark, and Bob Coecke. 2014. Frobenius anatomy of word meanings 2: possessive relative pronouns. *Journal of Logic and Computation*, 26:785–815.
- G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.
- H. Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.
- Gijs Jasper Wijnholds. 2017. Coherent diagrammatic reasoning in compositional distributional semantics. In *Proc. 24th WoLLIC Workshop*, pages 371–386.