Polar Questions in SPA–TTR: Linking Dialogue, Acquisition, and Neurosemantics

Jonathan Ginzburg[1], Shiyun Dong[1], Staffan Larsson[2], Robin Cooper[2], Andy Lücking[3]
[1]Université Paris Cité, CNRS, [2]University of Gothenburg, [3] Technische Universität Chemnitz

Abstract

In this paper, we offer an extension of an earlier proposal for treating wh-questions within a compositional, neurally–implemented semantic framework that interfaces with memory to the other main class of questions, namely polar questions. Our proposal yields improved empirical coverage for polar questions as compared with previous formal semantic accounts. It also offers the basis for an account of the finding that understanding for wh-questions emerges in language development *before* that of polar questions–a finding that goes against all previous formal semantics accounts of questions where polar questions are simplest in terms of their semantic complexity.

1 Introduction

Formal semantics and cognitive neuroscience have not always been in tight collaboration. Frege (1918, e.g., p. 69) (the originator of compositionality) believed (scientific) analysis of meaning should be divorced from psychological considerations, whereas (Fodor, 1974) argues that appealing to 'lower-level' sciences is useless for understanding cognitive systems, giving rise to the criteria of (Fodor and Pylyshyn, 1988) for explanatory cognitive systems which includes compositionality.

Considerable progress in bringing semantics and cognitive neuroscience together has been achieved by work pioneered in the Semantic Pointer Architecture (SPA) (Eliasmith, 2013). This framework underpins what is currently the world's largest functional brain model (Spaun Eliasmith, 2013; Eliasmith et al., 2012; Voelker and Eliasmith, 2023), which includes perception, decision making, and motor control systems integrated in a cognitive model that is implemented in spiking neurons and captures detailed anatomical and physiological characteristics of the mammalian brain. The SPA's structured representations are a neural implementation of a Vector Symbolic Architecture (VSA; Gayler, 2004; Schlegel et al., 2022).

The basic strategy of the SPA is to couple a VSA's algebraic structure on a vector space, with coding and decoding operations into ensembles of neurons. In this way, one can both maintain compositional analyses of natural language in a transparent way that differs sharply from Large Language Model approaches, while retaining the robustness and the continuity vectors provide in semantic space.

While significant progress has been made within such an approach, it has dealt so far with relatively simple semantic phenomena. (Larsson et al., 2025) offers extension of this approach to a complex semantic phenomenon involving variable binding, developing an analysis of wh–questions. In contrast to existing formal semantic treatments, this analysis enables an interface with memory, since answering questions involves in some cases long term memory (e.g., (1a)), in others working memory, at times a combination thereof (e.g., (1b)).

(1) a. What is the largest city in Kurdistan?

b. Are you aware of the tsetse fly on your plate?

In this paper, we extend this approach to polar questions. We analyze a polar question $p?$ as a propositional function $\lambda P.P(p)$ that seeks to instantiate a modifier of the proposition p and show how this can be implemented in SPA. What emerges is a unified, compositional treatment of the two major classes of questions[1] in a neurosemantic framework. Beyond the explicit interface with memory, our analysis of polar questions is superior to existing accounts of polar questions developed in the formal semantics literature in two important ways:

[1]Evidence from 10 languages show these two classes constitute typically >95% of questions (Bolden et al., 2023).

Third Workshop on the Bridges and Gaps between Formal and Computational Linguistics, pages 40–49

1. **Answerhood**: Formal semantic treatments of polar-questions (with a few exceptions discussed below) analyze these as the simplest type of question, whose answers are 'yes' and 'no' (hence the term 'yes/no–questions). As we will demonstrate, this widely held assumption is contradicted by both introspective and corpus data that shows such questions give rise to an intrinsically wider range of answers, which our analysis captures.

2. **Order of emergence in language development**: recent data from language acquisition (Moradlou et al., 2021) shows that across languages, wh-questions are understood by infants *before* polar questions, despite biases favouring the latter in the input.[2] As Moradlou et al. (2021) demonstrate, (in the languages they consider and probably in general) polar questions are clearly not more complex syntactically than wh-questions. This suggests that polar questions are in some way more complex semantically than (certain types of) wh-questions, contradicting virtually all existing formal semantic analyses, as discussed below. Our analysis underpins the explanation offered by Moradlou et al. (2021) for the order of emergence, tied as it is to the relative difficulty of grounding answers of a given type of question to elements in the context. For the early understood wh-questions, these are concrete entities (*Carer: What is this?* (Carer points to the object): *A pencil*), whereas for polar questions these are (abstract) propositional modifiers (*Carer: Is the pencil green* Yes (It is green)/No (It is not green)).

The structure of the paper is as follows: Section 2 provides background on existing work on polar questions, the Semantic Pointer Architecture (SPA), and Type Theory with Records, the high level symbolic theory we use for semantic formulation. In Section 3 we describe our proposal for the analysis of polar questions. Section 4 reviews and extends the account of Larsson et al. (2025) of wh-questions in SPA to incorporate the analysis we propose of polar questions. In Section 5 we report the results we obtained of a simulation of our account within the neuro-simulator Nengo (Bekolay et al., 2014). In Section 6, we conclude and describe future work.

[2]This finding is based both on a corpus study (in the Providence Corpus (American English) (**?**) showing that children provide answers to certain wh–questions **before** 'yes'/'no' are produced. It is also based on experimental work involving elicitation studies (in German and Chinese).

2 Background

2.1 The semantics of polar questions

There are a wide range of approaches to the semantics of questions (see Wiśniewski, 2015 for a review.). Nonetheless, on just about all existing theories, the entities denoted by wh-interrogatives are more complex entities than those denoted by polar-interrogatives. For instance, in what is probably the most commonly held view, the approach of (Hamblin, 1973), polar interrogatives denote the set $\{p, \neg p\}$, whereas wh-questions denote a potentially large set of possible candidate entities that might instantiate the property in question; in the partition theory (Groenendijk and Stokhof, 1997), a polar-interrogative denotes a partition that has exactly two entities (corresponding to the question being resolved positively or negatively, respectively.) With a unary wh-interrogative, the size of the partition is less determinate, it depends on the number of possible candidate entities that could serve as fillers for the property in question—if that number is known to be n, then the size is 2^n.

Related to this is the fact that the notion of answerhood provided by such theories recognizes for a polar question p? two answers, p and its negation $\neg p$. The motivation for this is that when interrogatives are embedded by *resolutive* predicates (e.g., 'know' or 'discover') they coerce the interrogative to denote a proposition that resolves the question (Lahiri, 2002; Ginzburg and Sag, 2000) and for polar questions these are p and $\neg p$. As pointed out by (Ginzburg, 1995; Omar, 2020) there are a variety of other propositions that count as *about* a question, without potentially resolving it:

(2) a. Jill: Is Millie leaving tomorrow? Bill: Possibly/It's unlikely/If it doesn't rain.

b. Bill provided information about whether Millie is leaving tomorrow. ((Ginzburg, 1995), ex., (99,101))

Corpus studies indicate that such answers are not rare: in a study of polar answers in (a subcorpus of) the Switchboard corpus by (de Marneffe et al., 2009) non-polar answers constitute 21/638 responses, whereas in a study of indirect answers to

polar questions building subcorpora from Switchboard and the Circa corpora (Wang et al., 2024) provide figures of 55% (Movie domain), 7% (Travel domain), and 34.7% (Tennis domain) (n=900) for responses not entailing 'yes' or 'no' (though these presumably also include cases classified as 'evasive responses' (e.g., 'I don't know', 'Not a question I would like to answer etc) in the taxonomy of responses to questions of (Ginzburg et al., 2022).).

2.2 SPA (Semantic Pointer Architecture)

SPA utilizes *holographic reduced representations* (HRR; Plate, 2003). These are a particular implementation of compressed representations, that is, (higher-order) semantic representations that are obtained by "compressing" (lower-level) semantic representations (Hinton, 1990). HRRs achieve this with two key operations: vector addition that simulates conjunction, and *circular convolution*, a multiplication operation that binds high-dimensional vectors of dimension d into a new vector of dimension d. HRR is a true-to-dimension instance of a *Vector Symbolic Architecture* (VSA; Gayler, 2003), in contrast to, for instance, tensor products (Smolensky, 1990). This feature of HRR is crucially for biological plausibility in terms of space utilized for representation (Stewart and Eliasmith, 2012)

The vector algebra of HRR includes the following operators ($\mathbf{a}$, $\mathbf{b}$, ... are (normalized unit) vectors.):

- $+$: $\mathbf{a}+\mathbf{b} = [\mathbf{a}_0+\mathbf{b}_0, \mathbf{a}_1+\mathbf{b}_1, \ldots, \mathbf{a}_{d-1}+\mathbf{b}_{d-1}]$
- $-$: $\mathbf{a}-\mathbf{b} = [\mathbf{a}_0-\mathbf{b}_0, \mathbf{a}_1-\mathbf{b}_1, \ldots, \mathbf{a}_{d-1}-\mathbf{b}_{d-1}]$
- $\circledast$: $\mathbf{c} = \mathbf{a} \circledast \mathbf{b} : \mathbf{c}_j = \sum_{k=0}^{d-1} \mathbf{a}_k \mathbf{b}_{j-k \pmod d}$
- inverse: $\mathbf{a}' = [\mathbf{a}_0, \mathbf{a}_{d-1}, \mathbf{a}_{d-2}, \mathbf{a}_{d-3}, \ldots, \mathbf{a}_1]$

$+$ and $\circledast$ have many properties in common with both scalar and matrix multiplication. They are commutative, associative, and satisfy distributivity. In this approach, contents for NL expressions can be compositionally built up with potential recursion using these operations *on vectors*, as exemplified in (3); (here and below *dog*_52,*cat*_63, etc. are vectors corresponding to a fixed entity, or class thereof.)

(3) a. The dog chases the cat $\mapsto$ **chase**$\circledast$**rel**$+$**dog_52**$\circledast$**agent**$+$**cat_63**$\circledast$**theme**

b. John thinks the dog chases the cat $\mapsto$ **John_17** $\circledast$ **theme** $+$ **think** $\circledast$ **rel** $+$ (**chase**$\circledast$**rel**$+$**dog_52**$\circledast$**agent**$+$**cat_63**$\circledast$**theme**) $\circledast$ **proparg**

The fact that any vector **a** has an approximate inverse **a'** (for which **a** $\circledast$ **a'** ≈ 1) plays a key role in enabling a theory of questions to be developed. Given a structure such as

(4) **agent** $\circledast$ **dog**$_{52}$ $+$ **frame** $\circledast$ **chase** $+$ **theme** $\circledast$ **cat**$_{43}$

"Who does dog$_{52}$ chase?" amounts to unbinding (4) with **theme**$'$ and thereby retrieving an answer.

(5) (4) $\circledast$ **theme**$'$ $=$ **agent** $\circledast$ **dog**$_{52}$ $\circledast$ **theme**$'$ $+$ **frame** $\circledast$ **chase** $\circledast$ **theme**$'$ $+$ **theme** $\circledast$ **cat**$_{43}$ $\circledast$ **theme**$'$ = noise + **cat**$_{43}$ $\approx$ **cat**$_{43}$

It is noteworthy that the true-to-dimension HRR vector manipulations are lossy: in particular the inverse $\mathbf{a}'$ of a vector **a** is not the exact inverse but approximates it. This "lossiness" introduces the need for clean-up memories when using it in cognitive VSA architectures like the SPA (see below).

2.3 Type Theory with Records (TTR)

Our general strategy to semantic analysis is to formulate analyses in a high level (symbolic) semantic framework, here Type Theory with Records (TTR). (Cooper, 2023; Chatzikyriakidis et al., 2025). An important motivation for using TTR is that all complex TTR objects are constructed from labelled sets, which correspond to the representation of structured objects which SPA achieves using superposition and circular convolution. This is the basis for a systematic mapping to SPA (Larsson et al., 2023), which in turn has a direct mapping into neural networks via the *Neural Engineering Framework* (NEF; Eliasmith and Anderson, 2003) which implements vectorial representations and manipulations in neural simulations.[3] How many levels of explanation are necessary is an open question, but as Eliasmith and Kolbeck (2015, e.g., p. 331) demonstrate, it is often necessary to '[be] thinking about multiple levels simultaneously when constructing information-processing systems.'

[3] See https://www.nengo.ai/.

We sketch here those aspects of TTR that are used in this paper. $s : T$ represents a judgement that s is of type T. Types may be either *basic* or *complex* (in the sense that they are structured objects which have types or other objects introduced in the theory as components). One basic type that we will use is *Ind*, the type of individuals; another is *Real*, the type of real numbers.

Among the complex types are *ptypes* which are constructed from a predicate and arguments of appropriate types as specified for the predicate. Examples are 'woman(a)', 'see(a,b)' where a, b : *Ind*. The objects or *witnesses* of ptypes can be thought of as situations, states or events in the world which instantiate the type. Thus s : woman(a) can be glossed as "s is a situation which shows (or proves) that a is a woman".

Another kind of complex type are *record types*. In TTR *records* are modelled as a labelled set consisting of a finite set of fields. Each field is an ordered pair, $\langle \ell, o \rangle$, where ℓ is a *label* (drawn from a countably infinite stock of labels) and o is an object which is a witness of some type. No two fields of a record can contain the same label. Importantly, o can itself be a record.

A *record type* is like a record except that the fields are of the form $\langle \ell, T \rangle$ where ℓ is a label as before and T is a type. The basic intuition is that a record, r is a witness for a record type, T, just in case for each field, $\langle \ell_i, T_i \rangle$, in T there is a field, $\langle \ell_i, o_i \rangle$, in r where $o_i : T_i$. (Note that this allows for the record to have additional fields with labels not included in the fields of the record type.) The types within fields in record types may *depend* on objects which can be found in the record which is being tested as a witness for the record type. We use a graphical display to represent both records and record types where each line represents a field. Example (6(i)) represents the type of records which can be used to model situations where a woman runs, whereas (6(ii)) represents a record of that type where a : *Ind*, s : woman(a) and e : run(a):

(6) (i)
$$\begin{bmatrix} \text{ref} & : & \textit{Ind} \\ \text{c}_{\text{woman}} & : & \text{woman(ref)} \\ \text{c}_{\text{run}} & : & \text{run(ref)} \end{bmatrix}$$

(ii)
$$\begin{bmatrix} \text{ref} & = & a \\ \text{c}_{\text{woman}} & = & s \\ \text{c}_{\text{run}} & = & e \\ \dots & & \end{bmatrix}$$

3 Questions: Semantic analysis

We will assume a view of questions as propositional functions, a view apparently initiated by Ajdukiewicz (1926), developed significantly in Kubinski (1960), and subsequently shared and further developed by a number of different approaches, e.g., Krifka (2001). This view offers (a) a direct explanation of short answers to questions (Jacobson, 2016) —the short answer corresponds to the 'missing argument' and (b) is the basis for the most detailed account of answerhood (Ginzburg and Sag, 2000, pp. 129–149) that we are aware of.

The simplest way to implement a propositional abstract in TTR, which we adopt here, is to treat a question as being an entity as in (7a), that is, a function from objects of a particular type returning a type that can be used as a proposition. A 'who'–question is exemplified in (7b), a 'where'–question is exemplified in (7c), and a multiple wh-question is exemplified in (7d):

(7) a. $\lambda x{:}T_1 \,.\, T_2$ where T_1 and T_2 are types.

b. Who chased the cat $\mapsto$
$\lambda x : \textit{Ind} \,.\, \left[\text{c1} : \text{Chase}(x,\text{c})\right]$

c. Where is the cat $\mapsto$
$\lambda l : \textit{Loc} \,.\, \left[\text{c2} : \text{In}(l,\text{c})\right]$

d. Who chased whom $\mapsto$
$\lambda x : \textit{Ind}, y : \textit{Ind} \,.\, \left[\text{c1} : \text{Chase}(x, y)\right]$

In past work on questions in TTR questions were treated as 0-ary abstracts, implemented as abstraction over an empty set or as unrestricted abstraction. Here we propose to unify polar-questions with wh-questions and identify the 'missing element' for polar questions as a propositional modifier, drawing on the data in (2). This yields a content exemplified in (8).

(8) Did the dog chase the cat $\mapsto$
$\lambda P : \textit{PropMod} \,.\, \left[\text{c3} : P(\text{Chase(d,c)})\right]$

As a consequence, the space of simple answers to the question—the instantiations of the propositional abstract the question is assumed to denote– is broadened widely for a wide range of answers such as *probably*, *certainly*, *unlikely*, *certainly not*, and even conditional answers such as *if it does not rain* can be included. This also yields a particular simple view of what 'yes' and 'no' denote as short answers, which gives rise to a unified account

together with answers to wh–questions: 'Yes' on this view is the identity function, whereas 'No' is a function that is the negation function for positive polar questions and the identity for negative polar questions, given the data in (9):

(9) a. Yes $\mapsto \lambda p : Type.p$

b. No $\mapsto$ $\{\lambda p : Type.\neg p \;\; q$: positive pol-question
$\lambda p : Type.p \;\; q$: negative pol-question$\}$

c. A: Do you want coffee? B: No (I don't want coffee).

d. A: You don't want coffee? B: No (I don't want coffee).

On this view polar questions involve searching for propositional modifiers in contrast to 'who'-questions and 'where'-questions, which which involves searches for (human) individuals and locations, respectively. This offers the required underpinning for the developmental finding by (Moradlou et al., 2021, p. 175) discussed in section 1: '*wh-questions emerge earlier because the answer that can be provided to such questions in "training sessions" between carer and child is easier to ground perceptually than the abstract entities expressed by propositional answers required for polar-questions.*'

4 Integration of semantic analysis of questions and neural modeling

4.1 Question construction in SPA

This paper builds directly on the framework developed by Larsson et al. (2025), which proposes a hybrid semantic–neural account of question answering, integrating a type-theoretic analysis of questions with the Semantic Pointer Architecture (SPA).

A λ-structure is encoded in SPA as a structured semantic pointer comprising (i) a vector representing the ptype corresponding to the question body, (ii) a λ-path pointer indicating the role or field in the ptype where the abstracted value is to be inserted, and (iii) a type constraint on the abstracted variable:

(10) a. $\lambda v : Ind.\text{chase}(\text{dog}_{52}, v)$

b. $\mathbf{Q} = \begin{pmatrix} \textbf{lambdapath} \circledast \textbf{arg2}+ \\ \textbf{lambdatype} \circledast \textbf{Ind}+ \\ \textbf{body} \circledast \begin{pmatrix} \textbf{pred} \circledast \textbf{chase}+ \\ \textbf{arg1} \circledast \textbf{dog}_{52}+ \\ \textbf{arg2} \circledast \mathbf{I} \end{pmatrix} \end{pmatrix}$

with $\mathbf{I}$ the identity vector $\mathbf{I} = [1, 0, 0, \ldots]$, such that for any vector $\mathbf{x}$, $\mathbf{I} \circledast \mathbf{x} = \mathbf{x}$. This is a variant of de Bruijn indexing (de Bruijn, 1972), coding lambda terms without using variables but using paths to mark positions instead.

Question answering is modelled as a function application and memory retrieval over such representations. λ-application is approximated by vector symbolic operations that replace the abstracted component of the question-body ptype with an answer vector, yielding a fully specified ptype:

(11) $f(Q, A) = Q \circledast \textbf{body}' - Path + Path \circledast A$
where $Path = Q \circledast \textbf{lambdapath}'$

For the specific $Q = \mathbf{Q}$ given above and $A = \mathbf{A}$ given in (12a), $Path = Q \circledast \textbf{lambdapath}'$ evaluates (with some noise) to **arg2**, hence yielding, as shown in (12b) the desired answer:

(12) a. $\mathbf{A} = \textbf{cat}_{43}$

b. $\mathbf{P} = \begin{pmatrix} \textbf{pred} \circledast \textbf{chase}+ \\ \textbf{arg1} \circledast \textbf{dog}_{52}+ \\ \textbf{arg2} \circledast \mathbf{I} \end{pmatrix} - \textbf{arg2} + \textbf{arg2} \circledast \textbf{cat}_{43} \approx \begin{pmatrix} \textbf{pred} \circledast \textbf{chase}+ \\ \textbf{arg1} \circledast \textbf{dog}_{52}+ \\ \textbf{arg2} \circledast \textbf{cat}_{43} \end{pmatrix}$

Memory retrieval exploits the structural similarity between the question-body ptype and stored ptypes in associative memory, where candidate ptypes are retrieved via unbinding and clean-up operations. Lastly, the abstracted value is extracted by unbinding with the λ-path pointer. Type compatibility between questions, answers, and ptypes is enforced using vector-encoded type information and clean-up over a domain of types.

In extending the SPA–TTR framework from wh-questions to polar questions, the modifier is represented as a separate meta-propositional field. The new proposition is augmented with a modifier slot, and the original proposition itself becomes an argument.

Accordingly, a polar question abstracts over the modifier field inside the nested structure rather than over an unstructured response token. A *Yes* answer

corresponds to binding the modifier role to a value YES, while *No* corresponds to binding it to a negative value NO. This strategy cleanly separates propositional content from modification, supports short answers, and avoids destructive vector negation.

(13)

$$\mathbf{Q} = \begin{pmatrix} \mathbf{lambdapath} \circledast \mathbf{modifier}+ \\ \mathbf{lambdatype} \circledast \mathbf{modifiertype}+ \\ \mathbf{body} \circledast \begin{pmatrix} \mathbf{modifier} \circledast \mathbf{I}+ \\ \mathbf{arg1} \circledast \begin{pmatrix} \mathbf{pred} \circledast \mathbf{chase}+ \\ \mathbf{arg2} \circledast \mathbf{dog}+ \\ \mathbf{arg3} \circledast \mathbf{cat} \end{pmatrix} \end{pmatrix} \end{pmatrix}$$

For example, with an answer such as (14), we expect to obtain a ptype such as (15) by inputting **Q** and **A** into a SPA network for λ-application.

(14) $\mathbf{A} = \mathbf{Yes}$

(15) $$\mathbf{P} = \begin{pmatrix} \mathbf{modifier} \circledast \mathbf{Yes}+ \\ \mathbf{arg1} \circledast \begin{pmatrix} \mathbf{pred} \circledast \mathbf{chase}+ \\ \mathbf{arg2} \circledast \mathbf{dog}+ \\ \mathbf{arg3} \circledast \mathbf{cat} \end{pmatrix} \end{pmatrix}$$

Using the schema in (11), the computation above outputs the body of the question with the λ- abstracted variable replaced by the argument, yielding the form indicated in (15).

4.2 Answer extraction from a polar question ptype

So far, we have focused on how to construct a complete propositional type by combining the given question and answer. In naturalistic question answering, however, the more common situation is the converse: a question is given as input to the network, and an appropriate answer must be extracted from a proposition stored in memory a priori. In this setting, the network is assumed to have access to a ptype **P** encoding a proposition that resolves the question **Q**, and the task is to retrieve the corresponding answer value **A**.

In Larsson et al. (2025), a type checking mechanism is implemented via a reconstruction test: after a candidate proposition is retrieved from memory, the model attempts to recombine it with the question's $\lambda-$abstraction, and type compatibility is determined by whether the resulting vector can be successfully cleaned up to a proper ptype. Failure is taken to indicate a type mismatch. The present work adopted this mechanism and extended it to polar questions by replacing abstraction over individuals with abstraction over **modifier** values.

The answer-extraction process utilizes the same structural information encoded in the question representation that is used for λ application. In particular, the λ-path field of the question specifies the role or position within the ptype from which the answer is to be retrieved. Answer extraction can therefore be implemented by unbinding the stored ptype with the inverse of the λ-path pointer supplied by the question:

(16) $f_{qp}(Q, P) = P \circledast (Q \circledast \mathbf{lambdapath}')'$

Intuitively, the modifier is first unbound with **lambdapath** to recover the role corresponding to the abstracted argument. This role pointer is then inverted and used to unbind the stored ptype, yielding the filler associated with that role.

For the example discussed above, where the question is *Does dog chase cat?* and the stored ptype encodes the proposition (15), the computation proceeds as:

(17) $\mathbf{A} = f_{qp}(\mathbf{Q}, \mathbf{P}) = \mathbf{P} \circledast (\mathbf{Q} \circledast \mathbf{lambdapath}')' \approx \mathbf{P} \circledast (\mathbf{modifier})' \approx \mathbf{Yes}$

5 A proof-of-concept model

5.1 Neural simulation results

To see if λ-abstracted question answering is feasible with neural network models, we implemented the key steps of Section 4 in Nengo (https://www.nengo.ai/).The encoded proposition reservoir, from which an answer is to be found, consists of 5 propositions, represented by role-filler semantic pointers (vectors of 256 or 512 dimensions). The contents of the propositions are the following: P1 = *dog chases cat*, P2 = *cat does not like mouse*, P3 = *mouse sees dog*, P4 = *cow does not chase cat*, P5 = *dog likes cow*. The inputs to the network are three questions, posed one after the other: 1. *Does dog chase cat?* 2. *Does cat like mouse?* 3. *Does mouse see dog?* Answers corresponding to the questions can be found in propositions 1, 2, and 3, respectively. To obtain the answers from the

propositions, the networks unbind the body and the λ-path from the question, subtract the λ-path, and unbind both to retrieve the value bound with the propositional modifier (i.e., compare the resulting vector to the semantic pointers in clean-up memory).

In (Larsson et al., 2025), an intermediate memory clean-up step, which, by design, is to use an index of ptypes for searching the answer, was introduced to reduce noise. In its implementation, an auto-associative module was used to achieve the desired outcomes. This module is preserved. After the ptypes **P1** to **P5** unbind the body from **Q** (see "Memory Input"), the intermediate output semantic pointer is processed by an auto-associative memory clean-up to pin point the most similar ptype by a *winner takes all* mechanism (see "Memory Output").

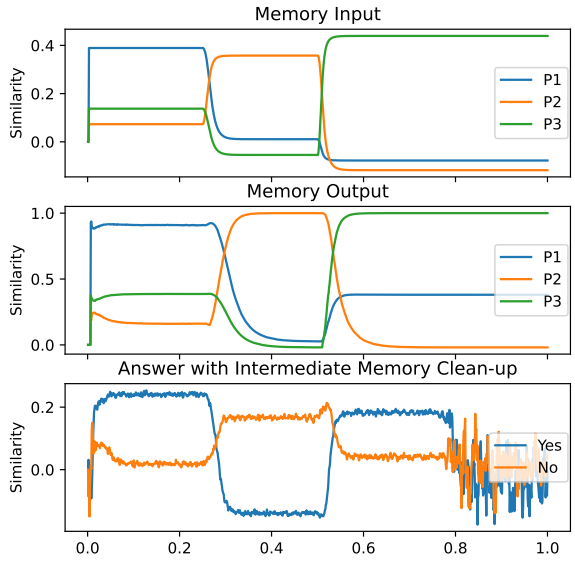

Figure 1: Retrieving a fragment answer according to 4.2 from a neural simulation. The neural simulation is dynamic because it runs in real time. The elapsed time (in seconds) is shown on the x-axis. The input to the network changes over time: from 0 to 0.25 the input is a semantic pointer corresponding to the question *Does dog chase cat?*, from 0.25 to 0.5 to the question *Does cat like mouse?*, and from 0.5 to 0.75 to *Does mouse see dog?*. From 0.75 onwards, no question is asked.

The simulation result is shown in the bottom row in Figure 1 ("Answer with Intermediate Memory Clean-up") and indicates the network's capacity to retrieve answers from encoded propositions via the λ-calculus. As is clear from the intermediate Figure 1 (which shows that winning P has similarity near 1 in all cases), the robustness issues arising from noise, exemplified in the bottom Figure 1 are not in relation to finding the right ptype but only in extracting out the polarity from the ptype once it has been identified. Thus, if the simulation instead looked solely for non-elliptical answers (answering with the whole ptype) we obtain near perfect performance. Robustness issues are likely due to unbinding from **P** with a noisy path ($\mathbf{Q} \circledast \mathbf{lambdapath}'$).

5.2 Robustness test results

Since representations and operations in the SPA rely on randomly initialized high-dimensional vectors, the behavior of the model is, in principle, sensitive to the choice of the initial state, i.e., a random seed. To assess the robustness of the proposed question-answer mechanism, a systematic evaluation was conducted. Specifically, we sampled 100 random seeds. For each seed, the full question-answering procedure was executed on the same set of 3 test questions. We used the number of correctly answered questions per trial, a score between 0 and 3, as the metric. 2 conditions were evaluated: 256- and 512-dimensions. As shown in Figure 2, performance is more stable at higher dimensionality: the 256-dimensional condition exhibits greater variability across random initializations, whereas the 512-dimensional condition shows a stronger concentration of perfect scores. This indicates that the proposed question-answering mechanism is robust to random initialization, more consistent at higher dimensionalities, and that human-like performance is not driven by idiosyncratic properties of specific random seeds.

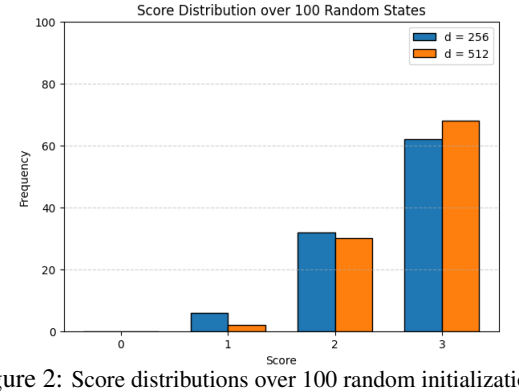

Figure 2: Score distributions over 100 random initializations for the polar question task. Each histogram shows the fraction of the three test questions answered correctly per trial for semantic pointer dimensionalities of 256 (blue) and 512 (orange). Higher dimensionality yields more consistent performance.

6 Concluding Remarks and Future Work

In this paper we offer an extension of an earlier proposal for treating wh-questions within a compositional, neurally–implemented semantic framework

that interfaces with memory to the other main class of questions, namely polar questions. Our proposal yields improved empirical coverage for polar questions as compared with previous formal semantic accounts. It also offers the basis for an account of the finding by (Moradlou et al., 2021) that understanding for wh-questions emerges in language development *before* that of polar questions—a finding that goes against previous accounts of questions in which polar questions are simplest in terms of their semantic complexity.

However, our account is still preliminary in a number of important respects that require developments in neural semantics and neural modelling of memory.

Answer Selection As mentioned in Section 4, the **modifier** term can contain different answers ranging from short answers to more complex conditional answers. Implementing such an enriched answer space in SPA would requires mechanisms beyond the binding and clean-up operations employed at current stage. In particular, distinguishing between graded or conditional answers plausibly depends on relations of entailment or compatibility between propositions, rather than on simple structural matching. Though SPA supports approximate similarity-based reasoning, modeling answerhood in this richer sense requires integrating explicit logical constraints, graded inference (Sadrzadeh et al., 2018), and probabilistic representations (Furlong et al., 2022) into our account. This is also necessary for developing answer selection strategies akin to exhaustivity.

Memory compartmentalization in SPA A central issue highlighted by the present work concerns the treatment of memory in SPA. The SPA we currently employ represents memory as a collection of semantic pointers in its vocabulary, without a principled architectural distinction between different memory systems such as long-term memory and its various subsystems (episodic, semantic, procedural etc) and working memory. Despite recent efforts to bridge this gap (Gosmann and Eliasmith, 2021), memory in SPA is largely modelled as a uniform representational substrate. From a computational perspective, this lack of compartmentalization limits the transparency. In the current model, propositional representations that function as background knowledge and those that are actively manipulated during question answering are treated uniformly. However, the operations involved in answering a question naturally decompose into stages with distinct functional roles: propositions must first be stored in a relatively stable form, plausibly corresponding to long-term or short-term memory, while the receipt of a question initiates a transient sequence of transformations that are more naturally associated with working memory.

Besides, a more articulated memory architecture is also desirable on linguistic and cognitive grounds. Empirical work suggests that language comprehension involves dynamic interaction between stable knowledge representations and temporarily maintained structures that are sensitive to the current conversational context (Daneman and Merikle, 1996). Treating all memory as a single homogeneous store risks conflating these distinct roles, whereas a whole network with distinct mechanisms might better reflect current views in psycholinguistics and cognitive neuroscience.

Acknowledgments

We thank three Brigap reviewers for their useful comments. This paper received support from Université Paris Cité as part of its program "Initiative d'excellence" IdEx (ANR-18-IDEX-0001) and from ANR CODIM.

References

Kazimierz Ajdukiewicz. 1926. Analiza semantyczna zdania pytajnego. *Ruch Filozoficzny*, 10:194b–195b.

Trevor Bekolay, James Bergstra, Eric Hunsberger, Travis DeWolf, Terrence C Stewart, Daniel Rasmussen, Xuan Choo, Aaron Voelker, and Chris Eliasmith. 2014. Nengo: a python tool for building large-scale functional brain models. *Frontiers in Neuroinformatics*, 7:48.

Galina B Bolden, John Heritage, and Marja-Leena Sorjonen. 2023. Chapter 1. introduction: Polar questions and their responses. *Responding to polar questions across languages and contexts*, pages 1–39.

Stergios Chatzikyriakidis, Robin Cooper, Eleni Gregoromichelaki, and Peter Sutton. 2025. *Types and the Structure of Meaning: Issues in Compositional and Lexical Semantics*. Cambridge Elements in Semantics. Cambridge University Press.

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

Meredyth Daneman and Philip M. Merikle. 1996. Working memory and language comprehension: A meta-analysis. *Psychonomic Bulletin & Review*, 3(4):422–433.

Nicolaas Govert de Bruijn. 1972. Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem. *Indagationes Mathematicae (Proceedings)*, 75(5):381–392.

Marie-Catherine de Marneffe, Scott Grimm, and Christopher Potts. 2009. Not a simple yes or no: Uncertainty in indirect answers. In *Proceedings of the SIGDIAL 2009 Conference*, pages 136–143.

Chris Eliasmith. 2013. *How to build a brain: A neural architecture for biological cognition*. OUP USA.

Chris Eliasmith and Charles H. Anderson. 2003. *Neural Engineering*. Computational Neuroscience. MIT Press, Cambridge, MA.

Chris Eliasmith and Carter Kolbeck. 2015. Marr's attacks: On reductionism and vagueness. *Topics in cognitive science*, 7(2):323–335.

Chris Eliasmith, Terrence C Stewart, Xuan Choo, Trevor Bekolay, Travis DeWolf, Yichuan Tang, and Daniel Rasmussen. 2012. A large-scale model of the functioning brain. *Science*, 338(6111):1202–1205.

Jerry A Fodor. 1974. Special sciences (or: The disunity of science as a working hypothesis). *Synthese*, pages 97–115.

Jerry A Fodor and Zenon W Pylyshyn. 1988. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71.

Gottlob Frege. 1918. Der Gedanke. *Beiträge zur Philosophie des deutschen Idealismus*, 1(2):58–77.

P Michael Furlong, Terrence C Stewart, and Chris Eliasmith. 2022. Fractional binding in vector symbolic representations for efficient mutual information exploration. In *Proceedings of the ICRA Workshop: Towards Curious Robots: Modern Approaches for Intrinsically-Motivated Intelligent Behavior*, pages 1–5.

Ross W. Gayler. 2003. Vector symbolic architectures answer Jackendoff's challenges for cognitive neuroscience. *ICCS/ASCS International Conference on Cognitive Science*, pages 133–138. Slightly updated 2004 in paper cs/0412059 on arXiv.

Ross W Gayler. 2004. Vector symbolic architectures answer jackendoff's challenges for cognitive neuroscience. *arXiv preprint cs/0412059*.

Jonathan Ginzburg. 1995. Resolving questions, i. *Linguistics and Philosophy*, 18:459–527.

Jonathan Ginzburg and Ivan A. Sag. 2000. *Interrogative Investigations: the form, meaning and use of English Interrogatives*. Number 123 in CSLI Lecture Notes. CSLI Publications, Stanford: California.

Jonathan Ginzburg, Zulipiye Yusupujiang, Chuyuan Li, Kexin Ren, Aleksandra Kucharska, and Paweł Łupkowski. 2022. Characterizing the response space of questions: data and theory. *Dialogue and Discourse*. https://journals.uic.edu/ojs/index.php/dad/article/download/11531/10757.

Jan Gosmann and Chris Eliasmith. 2021. Cue: A unified spiking neuron model of short-term and long-term memory. *Psychological Review*, 128(1):104.

Jeroen Groenendijk and Martin Stokhof. 1997. Questions. In Johan van Benthem and Alice ter Meulen, editors, *Handbook of Logic and Linguistics*. North Holland, Amsterdam.

C. L. Hamblin. 1973. Questions in montague english. In Barbara Partee, editor, *Montague Grammar*. Academic Press, New York.

Geoffrey E. Hinton. 1990. Mapping part-whole hierarchies into connectionist networks. *Artificial Intelligence*, 46(1):47–75.

Pauline Jacobson. 2016. The short answer: implications for direct compositionality (and vice versa). *Language*, 92(2):331–375.

Manfred Krifka. 2001. For a structured meaning account of questions and answers. *Audiatur Vox Sapientia. A Festschrift for Arnim von Stechow*, 52:287–319.

Tadeusz Kubinski. 1960. An essay in the logic of questions. In *Proceedings of the XIIth International Congress of Philosophy (Venetia 1958)*, volume 5, pages 315–322, Firenze. La Nuova Italia Editrice.

Utpal Lahiri. 2002. *Questions and Answers in Embedded Contexts*. Oxford University Press, Oxford.

Staffan Larsson, Robin Cooper, Jonathan Ginzburg, and Andy Luecking. 2023. Ttr at the spa: Relating type-theoretical semantics to neural semantic pointers. In *Proceedings of the 4th Natural Logic Meets Machine Learning Workshop*, pages 41–50.

Staffan Larsson, Jonathan Ginzburg, Robin Cooper, and Andy Lücking. 2025. Finding answers to questions: Bridging between type-based and computational neuroscience approaches. In *16th International Conference on Computational Semantics*, page 128.

Sara Moradlou, Xiaobei Zheng, TIAN Ye, and Jonathan Ginzburg. 2021. Wh-questions are understood before polar-questions: Evidence from english, german, and chinese. *Journal of Child Language*, 48(1):157–183.

AGHA Omar. 2020. Non-resolving responses to polar questions: A revision to the qud theory of relevance1. *Proceedings of Sinn und Bedeutung 24 Volume*.

Tony Plate. 2003. *Holographic Reduced Representation: distributed representation for cognitive structures*.

Mehrnoosh Sadrzadeh, Dimitri Kartsaklis, and Esma Balkır. 2018. Sentence entailment in compositional distributional semantics. *Annals of Mathematics and Artificial Intelligence*, 82(4):189–218.

Kenny Schlegel, Peer Neubert, and Peter Protzel. 2022. A comparison of vector symbolic architectures. *Artificial Intelligence Review*, 55(6):4523–4555.

Paul Smolensky. 1990. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, 46(1-2):159–216.

Terrence Stewart and Chris Eliasmith. 2012. Compositionality and biologically plausible models. In M. Werning, W. Hinzen, and E. Machery, editors, *The Oxford handbook of compositionality*, pages 596–615. Oxford University Press.

Aaron R Voelker and Chris Eliasmith. 2023. Programming with semantic pointers: A practical guide. *Journal of Cognitive Neuroscience*.

Zijie Wang, Farzana Rashid, and Eduardo Blanco. 2024. Interpreting answers to yes-no questions in dialogues from multiple domains. *arXiv preprint arXiv:2404.16262*.

Andrzej Wiśniewski. 2015. Semantics of questions. In *Handbook of Contemporary Semantic Theory, second edition*, Oxford. Blackwell.