A Frobenius Algebraic Analysis for Parasitic Gaps

Michael Moortgat
Utrecht University[*]
m.j.moortgat@uu.nl

Mehrnoosh Sadrzadeh
University College London[†]
m.sadrzadeh@ucl.ac.uk

Gijs Wijnholds
Utrecht University
g.j.wijnholds@uu.nl

Abstract

The interpretation of parasitic gaps is an ostensible case of non-linearity in natural language composition. Existing categorial analyses, both in the type-logical and in the combinatory traditions, rely on explicit forms of syntactic copying. We identify two types of parasitic gapping where the duplication of semantic content can be confined to the lexicon. Parasitic gaps in *adjuncts* are analysed as forms of generalized coordination with a polymorphic type schema for the head of the adjunct phrase. For parasitic gaps affecting *arguments* of the same predicate, the polymorphism is associated with the lexical item that introduces the primary gap. Our analysis is formulated in terms of Lambek calculus extended with structural control modalities. A compositional translation relates syntactic types and derivations to the interpreting compact closed category of finite dimensional vector spaces and linear maps with Frobenius algebras over it. When interpreted over the necessary semantic spaces, the Frobenius algebras provide the tools to model the proposed instances of lexical polymorphism.

[*]The research of the alphabetically first and third author is supported by NWO grant 360-89-070 “A composition calculus for vector-based semantic modelling with a localization for Dutch”.

[†]The alphabetically second author acknowledges the support of Royal Society International Exchange Award IE161631.

1 Introduction

Natural languages present many situations where an overt syntactic element provides the semantic content for one or more occurrences of elements that are not physically realized, or that have no meaning of their own. Illustrative cases can be found at the sentence and at the discourse level, e.g. long-distance dependencies in 'movement' constructions, ellipsis phenomena, anaphora. Parasitic gaps are a challenging case in point.

To provide the reader with the necessary linguistic background, the examples in (1) illustrate some relevant patterns[1]. The symbol $\sqcup$ marks the position of the virtual elements that depend on a physically realized phrase elsewhere in their context for their interpretation; in the generative grammar literature, these virtual elements are referred to as "gaps".

a papers that Bob rejected $\sqcup$ (immediately)
b Bob left the room without closing the window
c *window that Bob left the room without closing $\sqcup$
d papers that Bob rejected $\sqcup$ without reading $\sqcup_p$ (carefully)
e security breach that a report about $\sqcup_p$ in the NYT made $\sqcup$ public
f this is a candidate whom I would persuade every friend of $\sqcup$ to vote for $\sqcup$

(1)

Consider first the case of object relativisation in (a). This example has a single gap for the unexpressed direct object of *rejected*. In categorial type logics, gaps have the status of *hypotheses*, introduced by a higher-order type. In Lambek's [6] Syntactic Calculus, for example, the relative pronoun *that* in (a) would be typed as $(n\backslash n)/(s/np)$. The complete relative clause then acts as a noun postmodifier $n\backslash n$. The relative clause body *Bob rejected* $\sqcup$ is typed as s/np, which means it needs a noun phrase hypothesis in order to compose a full sentence. Because the hypothesis occupies the direct object position, it is impossible to physically realize that object, as the ungrammaticality of **papers that Bob rejected the proposal* shows. The Lambek type requires the hypothetical np to occur at the right periphery of the relative clause body — a restriction that we will lift in Section §2 to allow for phrase-internal hypotheses. An example would be (a) with an extra temporal adverb (*immediately*) at the end.

As the name suggests, a *parasitic* gap is felicitous only in the presence of a primary gap. The relative clause in (d) has two gaps: the primary one is for the

[1]For a more thorough discussion of the phenomena, and proposed analyses in a variety of grammatical frameworks, see[2].

object of *rejected* as in (a); the secondary, parasitic gap (marked by $\sqcup_p$) is the unexpressed object of *reading*. The parasitic gap occurs here in an adjunct: the verb phrase modifier *without closing* $\sqcup$. Such an adjunct by itself, is an *island* for extraction: the ungrammatical (c) shows that it is impossible for the relative pronoun to establish communication with a *np* hypothesis occuring within the adjunct phrase. Compare (c) with the gapless (b) which has the complete adjunct *without closing (the window)*$_{np}$.

Examples (e) and (f) represent a different type of parasitic gapping where both the primary and the parasitic gap regard co-arguments of the same verb. In (e), the primary gap is the direct object of *made public*, the secondary gap occurs in the subject argument of this predicate. In (f), the primary gap is the object of the infinitive complement of the verb *persuade*, viz. *to vote for* $\sqcup$, while the secondary gap occurs in the direct object of *persuade*[2].

We illustrated the adjunct and co-argument types of parasitic gapping in (1) with relative clause examples. Primary gaps can also be triggered in main or subordinate constituent question constructions, as in ($2a, b$), where *which papers* will carry the higher-order type initiating hypothetical reasoning. In the 'passive infinitive' case ($2c$), the higher-order type is associated with the adjective *hard*, which in this context could be typed as $ap/(to\_inf/np)$. The adjective then selects for an incomplete *to*-infinitive missing a *np* hypothesis, the direct object in ($2c$). As with the relative clause example ($1a$), putting a physically realized *np* in the position occupied by the hypothesis leads to ungrammaticality. Again, as in (1), the primary gaps here open the possibility for parasitic gaps dependent on them as in ($2d, e, f$). These examples also illustrate some of the various forms the adjunct phrase can take: temporal modification (*before, after*), contrastive (*despite*), etc.

- a which papers did Bob reject $\sqcup$ (immediately)
- b I know which papers Bob will reject $\sqcup$ (immediately)
- c this paper is hard to understand $\sqcup$ / *the proposal
- d which papers did Bob accept $\sqcup$ *despite* not liking $\sqcup_p$ (really)
- e I know which papers Bob will reject $\sqcup$ *before* even reading $\sqcup_p$ (cursorily)
- f this paper is easy to explain $\sqcup$ well *after* studying $\sqcup_p$ (thoroughly)

(2)

To account for the duplication of semantic content in parasitic gap constructions, existing categorial analyses rely on explicit forms of syntactic copying. The CCG analysis of [19] rests on (a directional version of) the **S** combinator of Combinatory Logic; the type-logical account of [13, 14] adapts the ! modality of Linear Logic to

[2] According to [19], each of the gaps in this type of example would be felicitous by itself.

implement a restricted form of the structural rule of Contraction. These syntactic devices are hard to control: the CCG version of the **S** rule is constrained by rule features; the attempts to properly constrain Contraction easily lead to undecidabilty as shown in [4].

Our aim in this paper is to explore *lexical polymorphism* as an alternative to syntactic copying. The technique of polymorphic typing is standardly used in categorial grammars for chameleon words such as *and, but.* Rather than giving these words a single type, they are assigned a type *schema*, with different realizations depending on whether they are conjoining sentences, verb phrases, transitive verbs, etc. Treating the adjunct phrases of $(1d)$ and $(2d, e, f)$ as forms of *subordinating* conjunction, we propose to similarly handle the adjunct type of parasitic gaps by means of a polymorphic type schema for the heads *without, despite, after*, etc. In the co-argument type of parasitic gapping $(1e{,}f)$, a conjunctive interpretation is absent. In this case, a polymorphic type schema for the relative pronouns *that* or *who(m)* allows us to generalize from the single gap instance $(1a)$ to the multi-gap case $(1e)$. To obtain the derived relative pronoun type from the basic assignment, we can rely on the same mechanisms that relate the basic type for *without* etc to the derived type needed for the parasitic gap examples.

Our analysis builds on the categorical Frobenius algebraic compositional distributional semantics of [16, 17], combined with a multimodal extension of Lambek calculus as the syntactic front end, as in [9]. Our analysis provides further evidence that Frobenius algebra is a powerful tool to model the internal dynamics of lexical semantics.

2 Syntax

2.1 The logic $\mathbf{NL}_\diamondsuit$

The syntactic front end for our analysis is the type logic $\mathbf{NL}_\diamondsuit$ of [10] which extends Lambek's pure logic of residuation [7] with modalities for structural control. The formula language is given by the following grammar (p atomic):

$$A, B ::= p \mid A \otimes B \mid A/B \mid A\backslash B \mid \diamondsuit A \mid \Box A \tag{3}$$

In $\mathbf{NL}_\diamondsuit$, types are assigned to *phrases*, not to strings as in the more familiar Syntactic Calculus of [6], or its pregroup version [8]. The tensor product $\otimes$ then is a non-associative, non-commutative operation for putting phrases together; it has adjoints / and \ expressing right and left incompleteness with respect to phrasal composition, as captured by the residuation inferences (4). In addition to the binary family $/, \otimes, \backslash$, the extended language has unary control modalities $\diamondsuit, \Box$ which again form a residuated pair with the inferences in (5).

$$A \longrightarrow C/B \text{ iff } A \otimes B \longrightarrow C \text{ iff } B \longrightarrow A \backslash C \tag{4}$$

$$\Diamond A \longrightarrow B \text{ iff } A \longrightarrow \Box B \tag{5}$$

The modalities serve a double purpose, either *licensing* reordering or restructuring that would otherwise be forbidden, or *blocking* structural operations that otherwise would be applicable. To license rightward extraction, as found in English long-range dependencies, we use the postulates in (6). Postulate $\alpha_\diamond$ is a controlled form of associativity: the $\Diamond$ marking licenses a rotation of the tensor formula tree that leaves the order of the components $A, B, \Diamond C$ unaffected. Postulate $\sigma_\diamond$ implements a form of controlled commutativity: here the internal structure of the tensor formula tree is unaffected, but the components B and $\Diamond C$ are exchanged.

$$\begin{array}{ll} \alpha_\diamond : & (A \otimes B) \otimes \Diamond C \longrightarrow A \otimes (B \otimes \Diamond C) \\ \sigma_\diamond : & (A \otimes B) \otimes \Diamond C \longrightarrow (A \otimes \Diamond C) \otimes B \end{array} \tag{6}$$

To block these structural operations from applying, we use a pair of modalities $\Diamond, \Box$. Phrases that qualify as syntactic islands are marked off by $\Diamond$. The modal island demarcation makes sure that the input conditions for $\alpha_\diamond, \sigma_\diamond$ do not arise. The island markers $\Diamond, \Box$ have no associated structural rules; their logical behaviour is fully characterized by (5).

NL$_\diamond$ derivations will be represented using the axiomatisation of Figure 1, due to Došen [3]. This axiomatisation takes (Co)Evaluation as primitive arrows, and recursively generalizes these by means of Monotonicity. It is routine to show that the residuation inferences of (4) and (5) become derivable rules given the axiomatisation of Figure 1. To streamline derivations, we will make use of the derived residuation steps. Also, we will freely use (Co)Evaluation and the structural postulates (6) in their *rule* form, by composing them with Transitivity ($\circ$).

2.2 Graphical calculus for NL$_\Diamond$

Wijnholds [23] gives a coherent diagrammatic language for the non-associative Lambek Calculus **NL**; the generalisation to **NL** with control modalities is straightforward, see Figure 2. In short, each connective is assigned two *links* that either compose or decompose a type built with that connective. Links (and diagrams) can be put together granted that their in- and outputs coincide. This system has a full recursive definition, and is shown to be sound and complete (i.e. coherent) with respect to the categorical formulation of the Lambek Calculus, given a suitable set of graphical equalities (not discussed in the current paper).

$$\frac{}{1_A : A \longrightarrow A} \qquad \frac{f : A \longrightarrow B \quad g : B \longrightarrow C}{g \circ f : A \longrightarrow C}$$

$$\frac{f : A \longrightarrow B \quad g : C \longrightarrow D}{f \otimes g : A \otimes C \longrightarrow B \otimes D}$$

$$\frac{f : A \longrightarrow B \quad g : C \longrightarrow D}{f/g : A/D \longrightarrow B/C} \qquad \frac{f : A \longrightarrow B \quad g : C \longrightarrow D}{f\backslash g : B\backslash C \longrightarrow A\backslash D}$$

$$\frac{f : A \longrightarrow B}{\Diamond f : \Diamond A \longrightarrow \Diamond B} \qquad \frac{f : A \longrightarrow B}{\Box f : \Box A \longrightarrow \Box B}$$

$$ev^{\backslash}_{A,B} : A \otimes A\backslash B \longrightarrow B \qquad co\text{-}ev^{\backslash}_{A,B} : B \longrightarrow A\backslash(A \otimes B)$$

$$ev^{/}_{A,B} : B/A \otimes A \longrightarrow B \qquad co\text{-}ev^{/}_{A,B} : B \longrightarrow (B \otimes A)/A$$

$$ev^{\Box}_{A} : \Diamond\Box A \longrightarrow A \qquad co\text{-}ev^{\Box}_{A} : A \longrightarrow \Box\Diamond A$$

$$\alpha_{\Diamond} : (A \otimes B) \otimes \Diamond C \longrightarrow A \otimes (B \otimes \Diamond C) \qquad \sigma_{\Diamond} : (A \otimes B) \otimes \Diamond C \longrightarrow (A \otimes \Diamond C) \otimes B$$

Figure 1: Došen style axiomatisation of $\mathbf{NL}_{\Diamond}$.

As an illustration, we present the derivation of the simple relative clause example (1*a*) in symbolic and diagrammatic form. For this case of non-subject[3] relativisation, the relative pronoun *that* is typed as a functor that produces a noun modifier $n\backslash n$ in combination with a sentence that contains an unexpressed np hypothesis (*Bob rejected* ␣ *immediately*). The subtype for the gap is the modally decorated formula $\Diamond\Box np$. The $\Diamond$ marking allows it to cross phrase boundaries on its way to the phrase-internal position adjacent to the transitive verb *rejected*. At that point, the licensing $\Diamond$ has done its work, and can be disposed of by means of the $ev^{\Box}$ axiom $\Diamond\Box np \longrightarrow np$, which provides the np object required by the transitive verb *rejected*. For legibility, we use words instead of their types for the lexical assumptions in the derivation below. The steps labeled ℓ indicate the lexical look-up.

[3]Subject relative clauses, e.g. *paper that* ␣ *irritates Bob*, do not involve any structural reasoning. The relative pronoun for subject relatives can be typed simply as $(n\backslash n)/(np\backslash s)$.

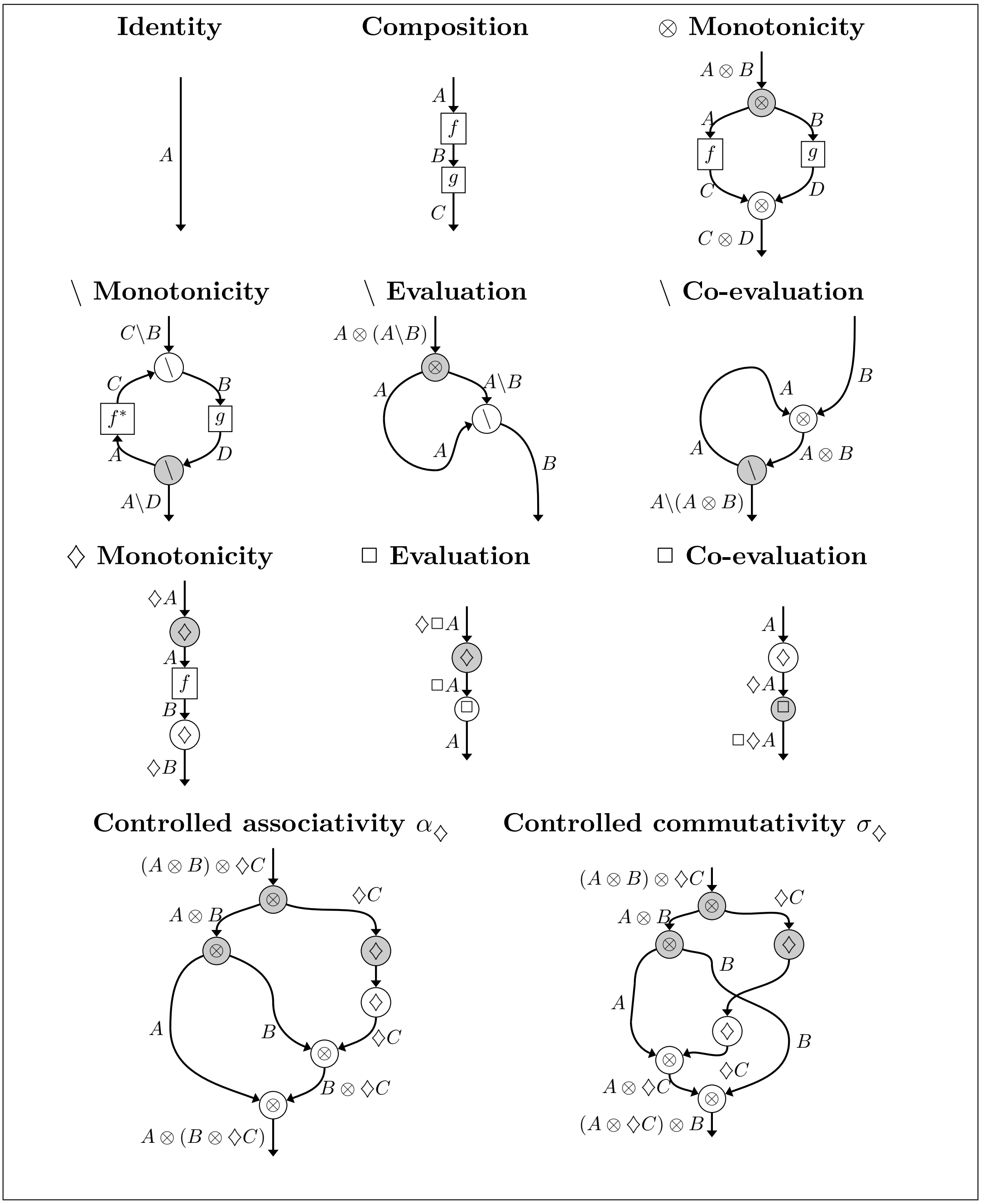


Figure 2: Došen style axiomisation of $\mathbf{NL}_\Diamond$ with diagrams. Monotonicity and (co)evaluation laws for / are fully symmetrical to the given diagrams for \.

$$
\dfrac{\dfrac{\text{paper}}{n}\,\ell \qquad \dfrac{\dfrac{\text{that}}{(n\backslash n)/(s/\Diamond\Box np)}\,\ell \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{Bob}}{np}\,\ell \quad \dfrac{\dfrac{\dfrac{\text{rejected}}{(np\backslash s)/np}\,\ell \quad \dfrac{}{\Diamond\Box np \longrightarrow np}\,ev^{\Box}}{\text{rejected}\otimes\Diamond\Box np \longrightarrow np\backslash s}\,ev^{/} \quad \dfrac{\text{immediately}}{(np\backslash s)\backslash(np\backslash s)}\,\ell}{(\text{rejected}\otimes\Diamond\Box np)\otimes\text{immediately}\longrightarrow np\backslash s}\,ev^{\backslash}}{\text{Bob}\otimes((\text{rejected}\otimes\Diamond\Box np)\otimes\text{immediately})\longrightarrow s}\,ev^{\backslash}}{\text{Bob}\otimes((\text{rejected}\otimes\text{immediately})\otimes\Diamond\Box np)\longrightarrow s}\,\sigma_{\Diamond}}{(\text{Bob}\otimes(\text{rejected}\otimes\text{immediately}))\otimes\Diamond\Box np\longrightarrow s}\,\alpha_{\Diamond}}{\text{Bob}\otimes(\text{rejected}\otimes\text{immediately})\longrightarrow s/\Diamond\Box np}\,res_{/}}{\text{that}\otimes(\text{Bob}\otimes(\text{rejected}\otimes\text{immediately}))\longrightarrow n\backslash n}\,ev^{/}}{\text{paper}\otimes(\text{that}\otimes(\text{Bob}\otimes(\text{rejected}\otimes\text{immediately})))\longrightarrow n}\,ev^{\backslash} \tag{7}
$$

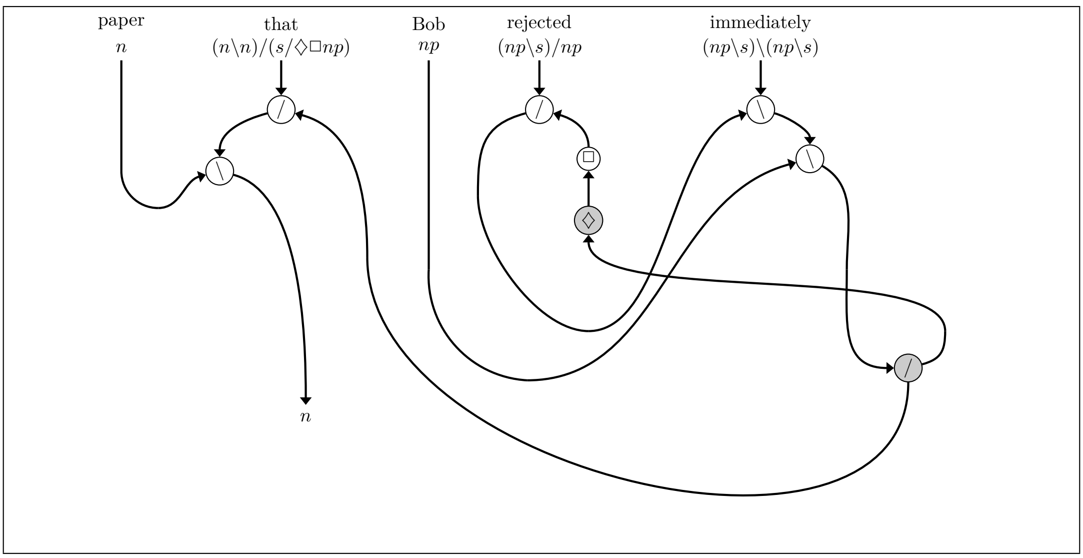


Figure 3: Diagrammatic form of *Paper that Bob rejected immediately.*

In the diagrammatic form of Fig 3, the $\Diamond\Box np$ gap hypothesis is indicated by the corresponding links. The leading $\Diamond$ link licenses the crossing over to the object position of *rejected* by means of the $\sigma_{\Diamond}$ postulate of Fig 2. In what follows, we use diagrams for $\mathbf{NL}_{\Diamond}$ derivations because this format pictures the information flow in a simple and intuitive way.

2.3 Typing Parasitic Gaps

Lexical polymorphism: generalized coordination As our account of parasitic gaps in adjuncts treats the adjuncts as a form of subordinate conjunction, we briefly review how lexical polymorphism is used in the analysis of generalized coordination.

Chameleon words such as *and, but* cannot easily be typed monomorphically; given an initial type and interpretation, say $(s\backslash s)/s$ for sentence coordination, we'd like to be able to obtain derived types and interpretations for the coordination of (in)transitive verbs, as in $(8b, c)$, or for non-constituent coordination cases such as $(8d)$.

$$\begin{array}{ll} a & \text{(Alice sings)}_s \text{ and (Bob dances)}_s \\ b & \text{Alice (sings and dances)}_{np\backslash s} \\ c & \text{Bob (criticized and rejected)}_{(np\backslash s)/np} \text{ the paper} \\ d & \text{(Alice praised)}_{s/\Diamond\Box np} \text{ but (Bob criticized)}_{s/\Diamond\Box np} \text{ the paper} \end{array} \qquad (8)$$

Deriving the $(b–d)$ types from an initial $(s\backslash s)/s$ assignment, however, goes beyond linearity. The attempt in (9) to derive verb phrase coordination from sentence coordination requires a copying step to strongly distribute the final np abstraction over the two conjuncts.

$$\dfrac{\dfrac{\dfrac{\vdots}{(\boxed{np} \otimes np\backslash s) \otimes ((s\backslash s)/s \otimes (\boxed{np} \otimes np\backslash s) \longrightarrow s}}{\boxed{np} \otimes (np\backslash s \otimes ((s\backslash s)/s \otimes np\backslash s) \longrightarrow s}\ \textsf{Copy!}}{(s\backslash s)/s \longrightarrow ((np\backslash s)\backslash(np\backslash s))/(\boxed{np}\backslash s)} \qquad (9)$$

Partee and Rooth's [15] work on generalized coordination offers a method for replacing syntactic copying by lexical polymorphism. Coordinating expressions *and, but* get a polymorphic type assignment $(X\backslash X)/X$ where X is a conjoinable type. The set of conjoinable types CType forms a subset of the general set of types Type. CType is defined inductively[4]:

- $s \in$ CType;
- $A\backslash B, B/A \in$ CType if $B \in$ CType, $A \in$ Type

The type polymorphism comes with a generalized interpretation. We write $\sqcap^X$ (infix notation) for a coordinator of (semantic) type $X \to X \to X$.

[4]Partee and Rooth formulate this in terms of the semantic types obtained from the syntax-semantics homomorphism h, with $h(s) = t$ (the type of truth values), $h(np) = e$ (individuals) and $h(A\backslash B) = h(B/A) = h(A) \to h(B)$.

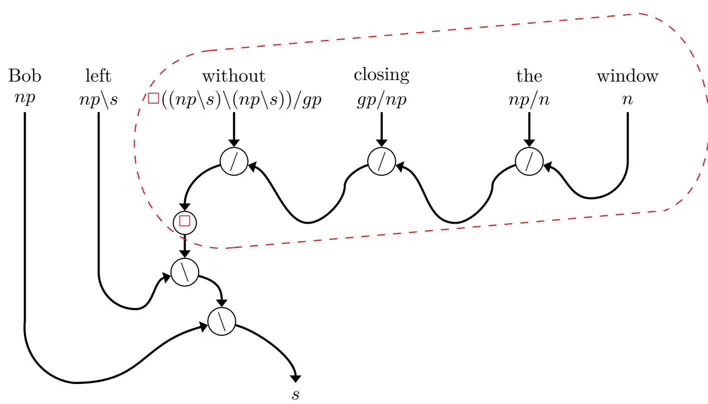
- $P \sqcap^t Q := P \wedge Q$ coordination in type t amounts to boolean conjunction
- $P \sqcap^{A \to B} Q := \lambda x^A.(P\ x) \sqcap^B (Q\ x)$ distributing the x^A parameter over the conjuncts

The generalized interpretation scheme, then, associates a type transition such as (9) with the Curry-Howard program that would be associated with a derivation involving the copying step. In Section §3, we will obtain the same effect using the Frobenius algebras over our vector-based interpretations.

Parasitic gaps in adjuncts Consider the type lexicon for the data in (1a–d)[5].

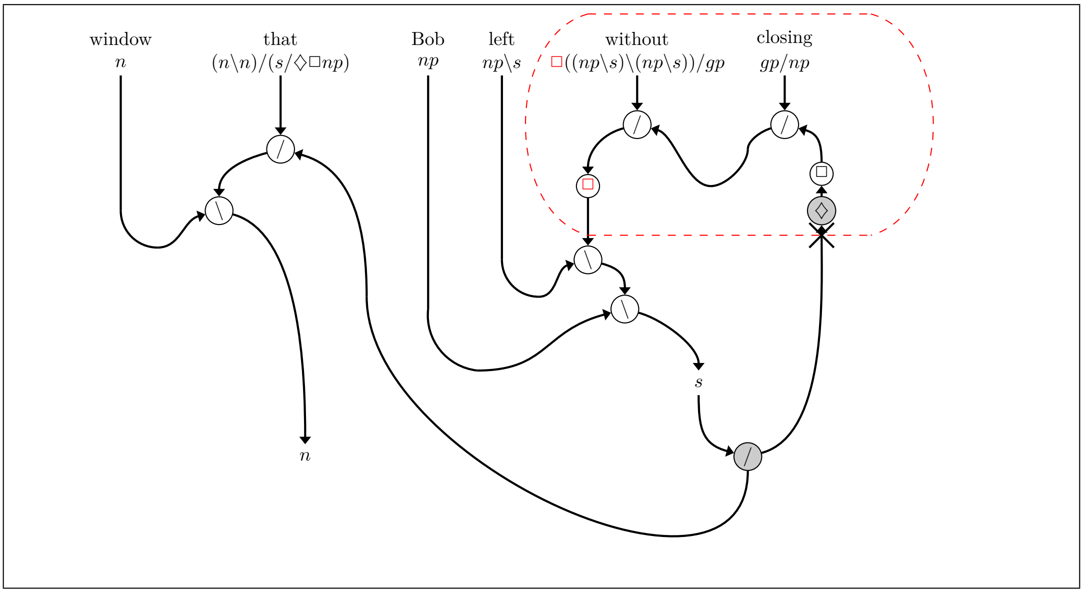
$$\begin{array}{rcl} \text{papers, window} & :: & n \\ \text{that} & :: & (n\backslash n)/(s/\Diamond\Box np) \\ \text{Bob} & :: & np \\ \text{rejected} & :: & (np\backslash s)/np \\ \text{reading, closing} & :: & gp/np \\ \text{immediately, carefully} & :: & iv\backslash iv \\ \text{without} & :: & \Box(X\backslash Y)/Z \qquad \text{(schematic)} \\ \text{without}^{b,c} & :: & \Box(iv\backslash iv)/gp \\ \text{without}^{d} & :: & \Box((iv/\Diamond\Box np)\backslash(iv/np))/(gp/\Diamond\Box np) \end{array} \tag{10}$$

The gap-less example (1b) provides the motivation for the basic type assignment to *without* as a functor combining with a non-finite gerund clause gp to produce a verb-phrase modifier $iv\backslash iv$. To impose island constraints, we use a pair of modalities $\Diamond, \Box$. In order to block the ungrammatical (1c), we follow [11] and lock the $iv\backslash iv$ result type with $\Box$; the matching $\Diamond$ needed to unlock it has the effect of demarcating the modifier phrase *without closing the window* as an island, represented in the diagram below by means of a dotted line.



[5] iv abbreviates $np\backslash s$; gp stands for gerund clause, headed by the -ing form of the verb.

An attempt to derive the ungrammatical *window that Bob left without closing* ␣ fails. The derivation proceeds like the one above, but with the gap hypothesis $\Diamond\Box np$ in the place of *the window*. At that point the $\Diamond$ island demarcation of *without closing* $\Diamond\Box np$ makes it impossible to bring out the hypothesis to the position where it can be withdrawn. This becomes apparent diagrammatically as the gap hypothesis cannot cross the dotted line:



Let us turn then to the adjunct parasitic gapping of (1*d*). To account for the double use of the gap we replace *syntactic* copying via controlled Contraction by *lexical* polymorphism, treating *without* as a polymorphic item on a par with coordinators *and, but*. That means we assign to *without* the following type schema

$$\text{without} :: \Box(X\backslash Y)/Z$$

with basic instantiation $X = Y = iv$, $Z = gp$. From this basic instantiation, a derived instantiation with $X = Y = iv/\Diamond\Box np$ and $Z = gp/\Diamond\Box np$ is obtained for the parasitic gapping example (1*d*) by uniformly dividing the subtypes iv and gp by $\Diamond\Box np$ using the forward slash.

In Section §3, we will see how the vector-based interpretation of the derived type is obtained in a systematic fashion from the interpretation of the basic type instantiation. For this, it is helpful to factorize the construction of the derived type as the combination of an expansion step and a distribution step. Ignoring the appropriate $\Box$ decoration to mark off the adjunct as an island, the expansion step here is an instance of the Geach transformation $A/B \longrightarrow (A/C)/(B/C)$, with $A = iv\backslash iv$, $B = gp$, $C = \Diamond\Box np$.

$$
\begin{array}{ll}
\text{(basic type)} & \Box(iv\backslash iv)/gp \\
& \downarrow \text{expand} \\
& (\Box(iv\backslash iv)/\Diamond\Box np)/(gp/\Diamond\Box np) \\
& \downarrow \text{distribute} \\
\text{(derived type)} & \Box((iv/\Diamond\Box np)\backslash(iv/\Diamond\Box np))/(gp/\Diamond\Box np)
\end{array}
$$

Setting now $A = iv$, $B = iv$, $C = \Diamond\Box np$, the distribution step is a directional instance of the **S** combinator $(A\backslash B)/C \longrightarrow (A/C)\backslash(B/C)$.

To arrive at the version of the derived type for *without* as we have it in our lexicon (10), a final calibration is required. We replace the result type $iv/\Diamond\Box np$ by iv/np, dropping the modal marking required for controlled associativity/commutativity. The final type $\Box((iv/\Diamond\Box np)\backslash(iv/np))/(gp/\Diamond\Box np)$ allows for the derivation of the parasitic gapping example (1c) displayed in Figure 4, but also for cases of Right Node Raising such as

Bob (rejected without reading)$_{iv/np}$ all papers about linguistics

where *all papers about linguistics* is a plain np rather than $\Diamond\Box np$.

Parasitic gaps: co-arguments

Let us turn to the co-argument type of parasitic gapping as exemplified by (1e, f). Consider first (1e), repeated here for convenience, together with a gap-less sentence that motivates the type-assignments given in (11).

security breach that a report about $\sqcup_p$ in the NYT made $\sqcup$ public $= (1e)$
(a report in the NYT)$_{np}$ made (the security breach)$_{np}$ public$_{ap}$

$$
\begin{array}{rcl}
\text{a, the} & :: & np/n \\
\text{security breach, report, NYT} & :: & n \\
\text{about, in} & :: & (n\backslash n)/np \\
\text{made} & :: & ((np\backslash s)/ap)/np \\
\text{public} & :: & ap
\end{array} \tag{11}
$$

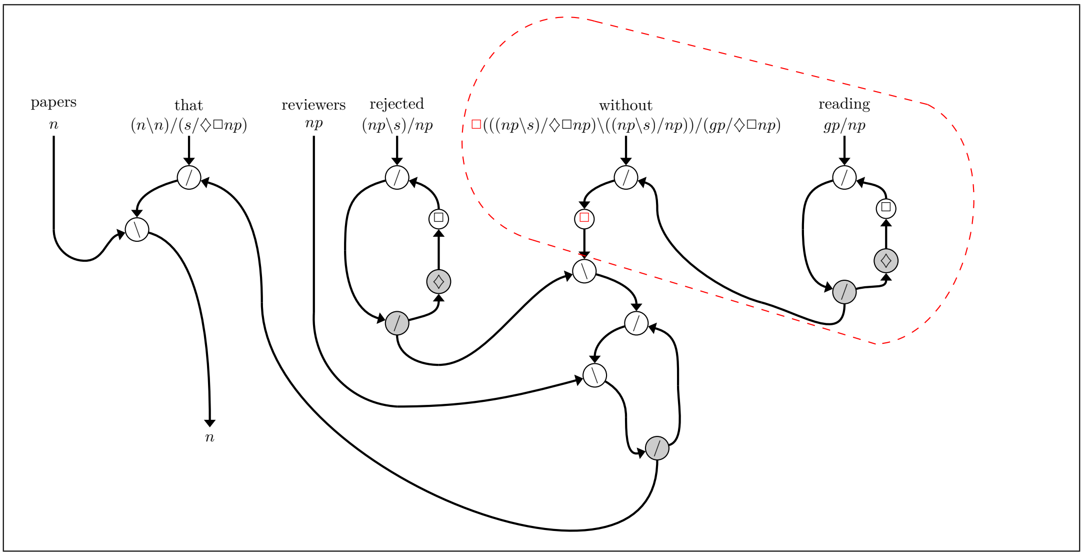


Figure 4: Information flow for the double parasitic gap.

In (1e) the relative clause body does not contain a coordination-like element that would be a suitable candidate to lexically encapsulate the ostensible copying. But we can turn to the relative pronoun itself, and use the mechanisms we relied on for parasitic gaps in adjuncts to move from the relative pronoun's basic type assignment for single-gap dependencies to a derived assignment for the double-gap dependency of (1e).

$$\begin{array}{rcl} \text{that}^{a,c} & :: & (n\backslash n)/(s/\Diamond\Box np) \\ \text{that}^{e} & :: & (n\backslash n)/((np/\Diamond\Box np)\otimes((np\backslash s)/\Diamond\Box np)) \end{array} \tag{12}$$

Again, we see that these types are derivable from the initial type for *that* by a combination of an expansion and a distribution step:

$$\begin{array}{c} (n\backslash n)/(s/\Diamond\Box np) \\ \Big\downarrow \text{expand} \\ (n\backslash n)/((np\otimes np\backslash s)/\Diamond\Box np) \\ \Big\downarrow \text{distribute} \\ (n\backslash n)/((np/\Diamond\Box np)\otimes((np\backslash s)/\Diamond\Box np)) \end{array}$$

The expansion step replaces s in antitone position by $np \otimes np\backslash s$, which is justified by leftward Application $ev^{\backslash} : np \otimes np\backslash s \longrightarrow s$ and Monotonicity. Here, with $A = np \otimes np\backslash s$, $B = s$, $C = \Diamond\Box np$ and $D = n\backslash n$, we have

$$\dfrac{\dfrac{\overline{A \longrightarrow B}\ \text{Appl}}{A/C \longrightarrow B/C}\ \text{Mon}^{\uparrow}}{D/(B/C) \longrightarrow D/(A/C)}\ \text{Mon}^{\downarrow}$$

Likewise, the distribution step relies on $\text{Mon}^{\downarrow}$ to replace $(A \otimes B)/C$ by $A/C \otimes B/C$ in antitone position. Here, with $A = np$, $B = np\backslash s$, $C = \Diamond\Box np$, $D = n\backslash n$, we have

$$\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{(A/C \otimes C) \otimes (B/C \otimes C) \longrightarrow A \otimes B}}{(A/C \otimes B/C) \otimes C \longrightarrow A \otimes B}\ \text{Distr}}{A/C \otimes B/C \longrightarrow (A \otimes B)/C}\ \text{Res}}{D/((A \otimes B)/C) \longrightarrow D/(A/C \otimes B/C)}\ \text{Mon}^{\downarrow}$$

Figure 6 has the derivation for example (1*e*).

Turning to (1*f*), repeated below with its underlying lexical type-assignments, we find the primary and secondary gaps in the infinitival complement *to_inf* and direct object of the verb *persuade*.

candidate whom Alice persuaded every friend of ␣ to vote for ␣ $\sim$ (1*f*)
Alice_{np} persuaded $(\text{a friend})_{np}$ to vote for Bob_{np}

$$\begin{array}{rcl} \text{persuaded} & :: & ((np\backslash s)/to\_inf)/np \\ \text{to vote} & :: & to\_inf/pp \\ \text{for} & :: & pp/np \\ \text{whom}^f & :: & (n\backslash n)/(((s/\Diamond\Box to\_inf)/\Diamond\Box np) \otimes (to\_inf/\Diamond\Box np)) \end{array} \quad (13)$$

To obtain the required derived type for *whom*, we follow the same expansion/distribution routine as for (1*f*). Expansion in this case replaces s by the product $(s/to\_inf) \otimes to\_inf$; the gap type $\Diamond\Box np$ is then distributed over the two factors of that product. To obtain the desired whom^f, there is an extra modal marking on the first occurrence of *to_inf*, in order to license rebracketing with respect to the subject. Recall that the base logic **NL** is non-associative by default.

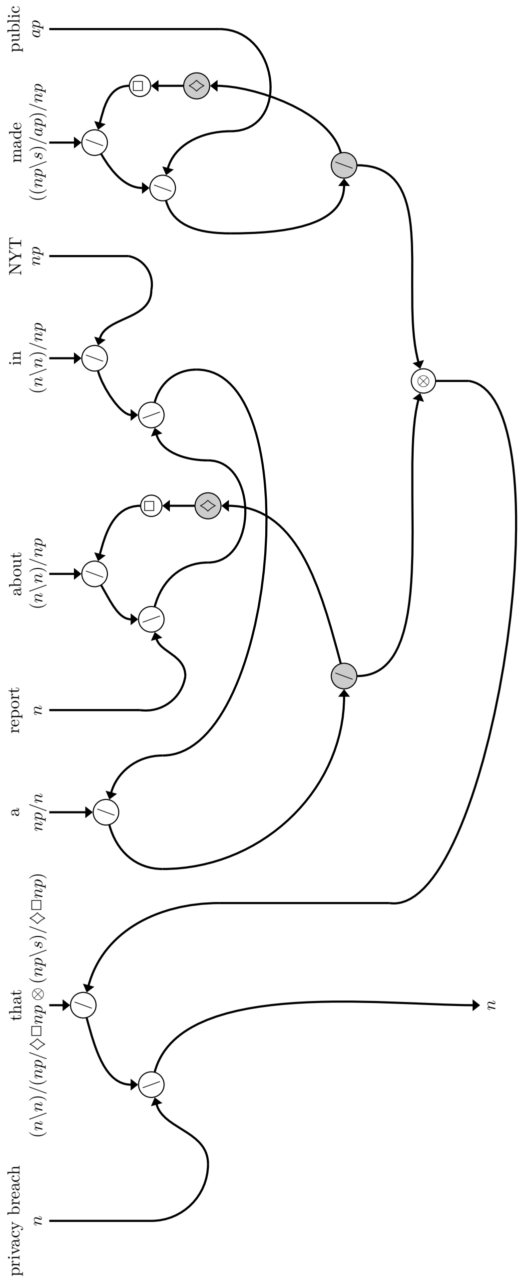


Figure 5: Co-argument parasitic gapping (1e).

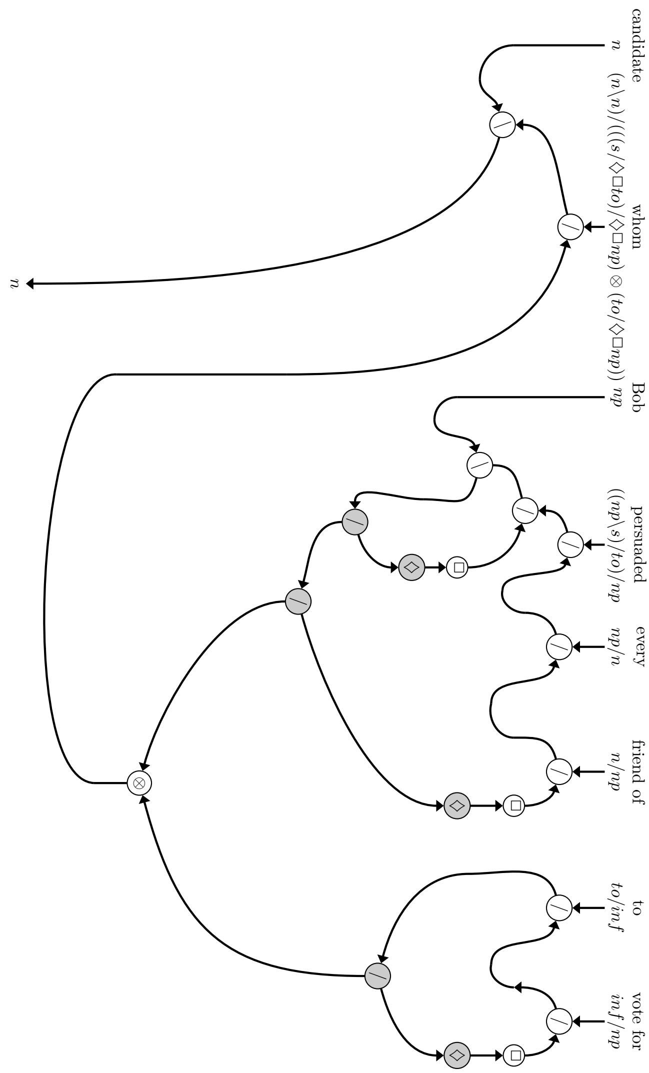


Figure 6: Co-argument parasitic gapping (1f). For brevity we write *to* for *to_inf*.

$$(n\backslash n)/(s/\Diamond\Box np)$$
$$\Big\downarrow \text{expand}$$
$$(n\backslash n)/((s/to\_inf)\otimes to\_inf)/\Diamond\Box np)$$
$$\Big\downarrow \text{distribute}$$
$$(n\backslash n)/(((s/to\_inf)/\Diamond\Box np)\otimes(to\_inf/\Diamond\Box np))$$

3 Frobenius Semantics

The proposed vector-based semantics has two ingredients: first, the *derivational* semantics specifies a compositional mapping that interprets types and proofs of the $\mathbf{NL}_\Diamond$ syntax as morphisms of a Compact Closed Category, concretely the category of **FVect** and linear maps. Second, the *lexical* semantics specifies the word-internal interpretation of individual lexical items; here, we make use of the Frobenius Algebras over **FVect** to model the copying of semantic content associated with the interpretation of relative pronouns such as *that* and *whom*, and modifier heads such as *without*.

3.1 Diagrams for Compact Closed Categories and Frobenius Algebras

Recall that a Compact Closed Category is a symmetric monoidal category $(\mathcal{C},\otimes,I)$ with duals A^* for every object A, and *contraction* and *expansion* maps for every object. In the case of vector spaces over fixed bases (our concrete semantics) we don't distinguish between objects and their duals, hence the contraction and expansion maps have signature $\epsilon : V\otimes V\to I$ and $\eta : I\to V\otimes V$, respectively.

For compact closed categories, there is a complete diagrammatic language available, that uses *cups* and *caps* to represent contraction and expansion, see [18]. These are drawn as connecting two objects either as a cup in the case of ϵ or as a cap in the case of η. The standard contraction and expansion maps of a CCC form the basis for interpreting derivations of $\mathbf{NL}_\Diamond$.

Crucial to our polymorphic approach is the inclusion of Frobenius Algebras in the lexicon. A Frobenius algebra in a symmetric monoidal category $(\mathcal{C},\otimes,I)$ is a tuple $(X,\Delta,\iota,\mu,\zeta)$ where, for X an object of $\mathcal{C}$, the first triple below is an internal comonoid and the second one is an internal monoid.

$$(X, \Delta, \iota) \qquad (X, \mu, \zeta)$$

This means that we have a coassociative map Δ and and its counit ι:

$$\Delta \colon X \to X \otimes X \qquad \iota \colon X \to I$$

and an associative map μ and its unit ζ:

$$\mu \colon X \otimes X \to X \qquad \zeta \colon I \to X$$

as morphisms of our category $\mathcal{C}$. The Δ and μ morphisms satisfy the *Frobenius condition* given below

$$(\mu \otimes 1_X) \circ (1_X \otimes \Delta) \;=\; \Delta \circ \mu \;=\; (1_X \otimes \mu) \circ (\Delta \otimes 1_X)$$

Informally, the comultiplication Δ decomposes the information contained in one object into two objects; the multiplication μ combines the information of two objects into one. In diagrammatic terms, to visualise the Frobenius operations one adds a white triangle to the diagrammatic language for CCCs that represents the (un)merging of information through the four different Frobenius maps. The resulting graphical language is summarised in Figure 7.

3.2 Derivational Semantics

For the derivational semantics, we need to define a homomorphism $\lceil \cdot \rceil$ that sends syntactic types and derivations to the corresponding components of the Compact Closed Category of **FVect** and linear maps. This homomorphism has been worked out by Moortgat and Wijnholds [9]. We present the key ingredients below and refer the reader to that paper for full details.
Types The target signature has atomic semantic spaces N and S, an involutive $(\cdot)^*$ for dual spaces and a symmetric monoidal product $\otimes$. We set

$$\begin{array}{rcl}
\lceil s \rceil & = & S, \\
\lceil np \rceil = \lceil n \rceil & = & N, \\
\lceil to_inf \rceil = \lceil ap \rceil = \lceil gp \rceil & = & N^* \otimes S, \\
\lceil \Diamond A \rceil = \lceil \Box A \rceil & = & \lceil A \rceil, \\
\lceil A/B \rceil & = & \lceil A \rceil \otimes \lceil B \rceil^*, \\
\lceil A \backslash B \rceil & = & \lceil A \rceil^* \otimes \lceil B \rceil
\end{array}$$

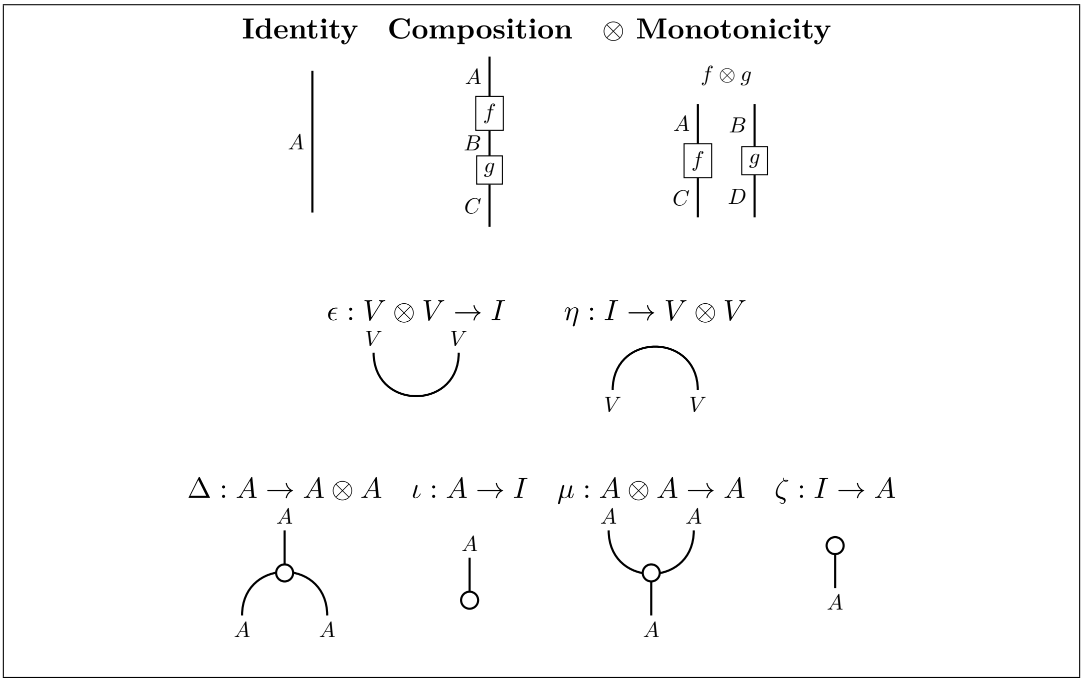


Figure 7: Diagrams of a Compact Closed Category with Frobenius Algebras.

Notice that *to_inf*, *ap* and *gp* are mapped to $N^* \otimes S$. Their understood subject is provided by the context: the main clause subject, in the case of *Bob fell asleep while watching TV*, the direct object in the case of *make the report public* and *persuade A to vote for B*.

Derivations The instances of the Evaluation axioms correspond to generalised contraction operations on vector spaces, the instances of the Co-Evaluation axioms dually are mapped to generalised expansion maps. The structural control postulates stipulate a syntactically limited associativity and commutativity; since the control modalities leave no trace on the semantic interpretation, the structural postulates $\alpha_\diamond$ and $\sigma_\diamond$ are interpreted using the standard associativity and symmetry maps of **FVect**.

The derivational semantics is represented graphically in Figure 8, where the diagrams of Figure 2 are interpreted in the complete diagrammatic language of compact closed categories of Figure 7.

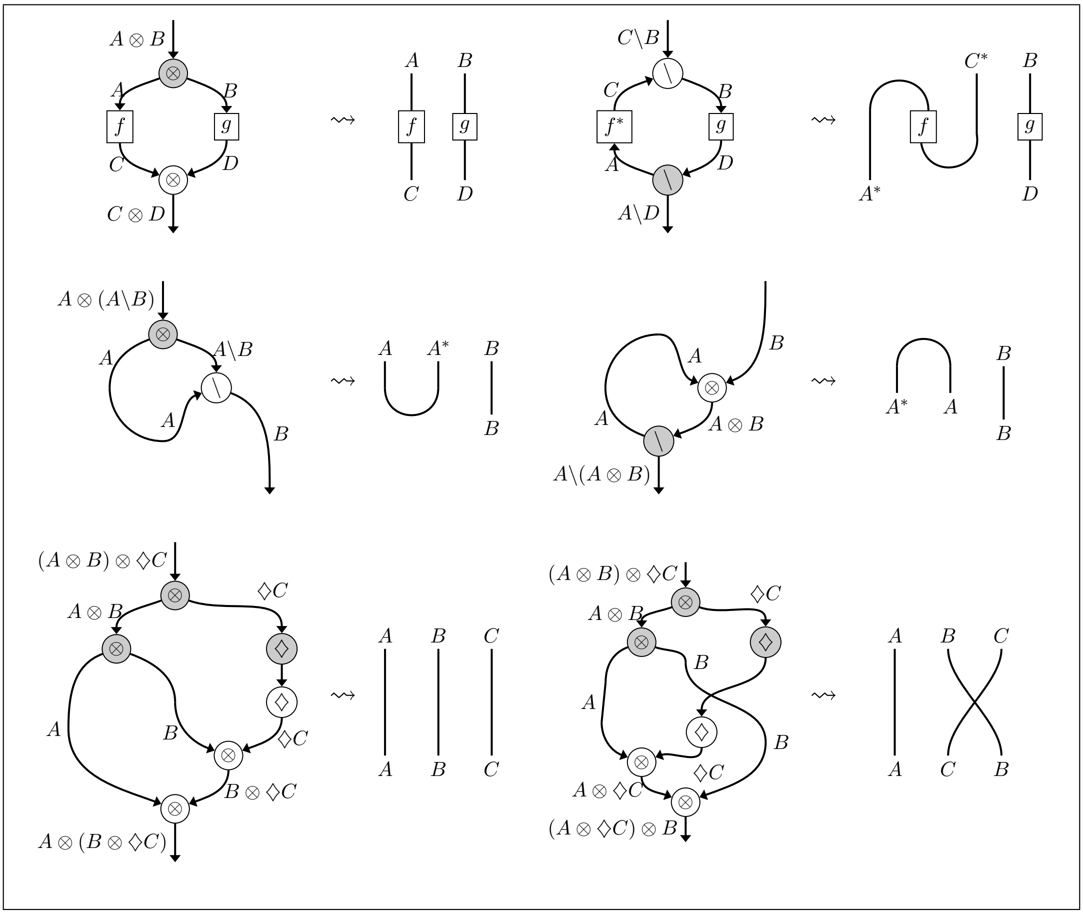


Figure 8: Interpreting derivations of $\mathbf{NL}_\Diamond$ arrows in a compact closed category.

Under the given interpretation, the diagrammatic derivation of Figure 4 for (1*d*)

papers	that	Bob	rejected	without	reading	
n	$(n\backslash n)/(s/\Diamond\Box np)$	np	$(np\backslash s)/np$	$(\Box(X\backslash Y))/Z$	gp/np	$\longrightarrow n$

is sent to the contractions in the interpreting CCC in Figure 9 (red: $\lceil$that$\rceil$, blue: $\lceil$without$\rceil$).

3.3 Lexical Semantics

For the lexical interpretation of the relative pronouns *that* and *whom* and the conjunctive *without*, we follow previous work [16, 17] and use Frobenius algebras that

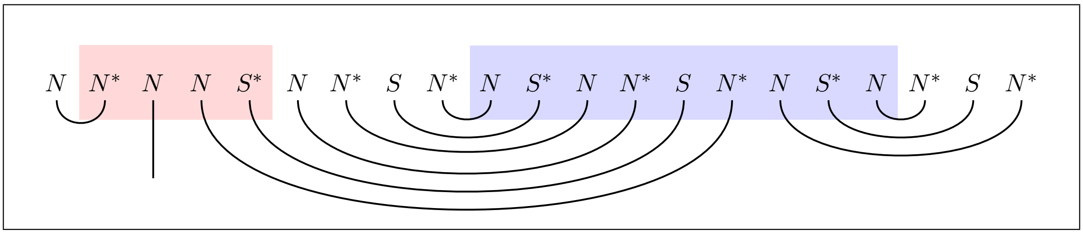


Figure 9: Axiom linking in a CCC for the parasitic gapping example (1d).

characterise vector space bases [1]. First, the basic form of the diagram for *that* is as developed in [16]. The basic diagram for *without* uses a double instance of a Frobenius Algebra to coordinate the gerundive phrase with the intransitive verb phrase consumed to its left. Recall that the interpretation homomorphism sends $np\backslash s$ and gp to the same semantic space, $N^* \otimes S$. In Figure 10 we display graphically these basic types as well as how their *derived* instantiations look. As our type for *whom* is derived similarly to the type of *that*, except that we distribute over the type to_inf rather than np, we get instead two extra wires rather than a single one.

For the case of parasitic gaps in adjunct positions we use the basic type for *that* and the derived type for *without*. For *that*, its basic Frobenius instantiation has the concrete effect of projecting down the verb phrase into a vector which is consecutively multiplied elementwise with the head noun of the main clause. The diagram for *without* then makes sure to distribute the missing hypothesis of the relative clause over the two gaps in the clause body. Given the identification $\lceil iv \rceil = \lceil gp \rceil$, this is essentially the treatment of coordination of [5].

For the co-argument case, we need make use of the derived type for *that*; its function is now to both specify the need for a clause body missing a hypothetical noun phrase, as well as coordinating this noun phrase through two gaps. Hence, the derived instantiation figures an iterative use of the Frobenius μ to merge three elements together.

With both the derivational semantics of Figure 9 and the lexical specifications of the constituents of Figure 10 we can put everything together to get the (unnormalised) diagram in Figure 11.

This diagram can be normalised under the equations of the diagrammatic language, leading to the normal form of Figure 12.

The above diagrams are morphisms of a symmetric compact closed category with Frobenius algebras and can be written down in that language as done e.g. in [16, 9]. Here, we provide the closed linear algebraic form of the normal form in Figure 12. For $\overline{\overline{\text{Rejected}}}$ and $\overline{\overline{\text{Not-Reading}}}$ the rank 3 tensors interpreting *rejected* and *(without)*

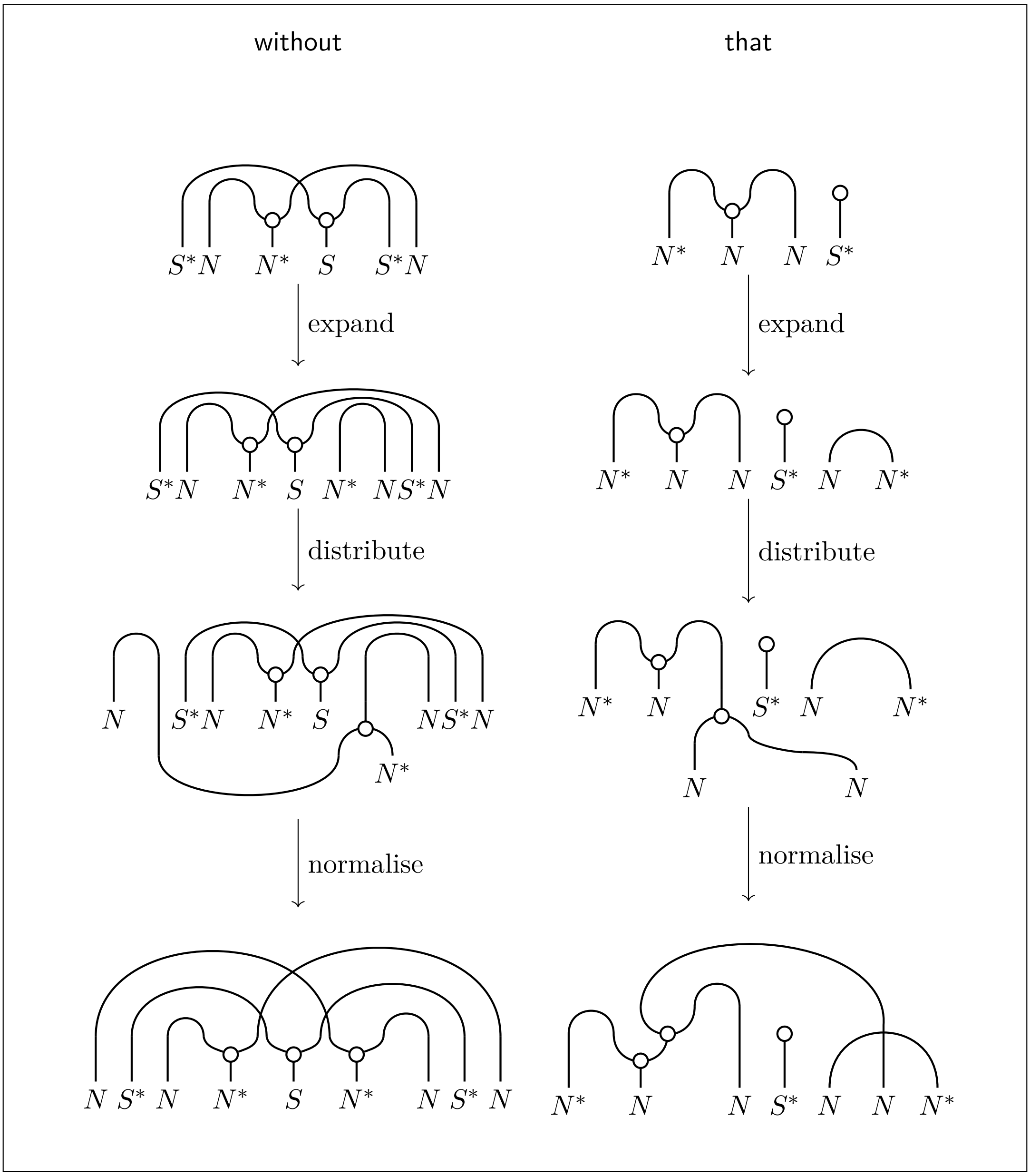


Figure 10: Deriving the lexical semantics for *without* and *that*.

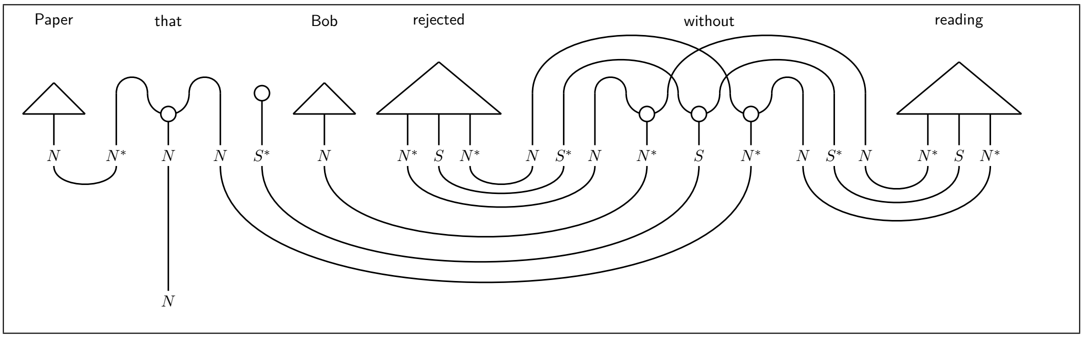


Figure 11: Semantic information flow for the double parasitic gap (initial form).

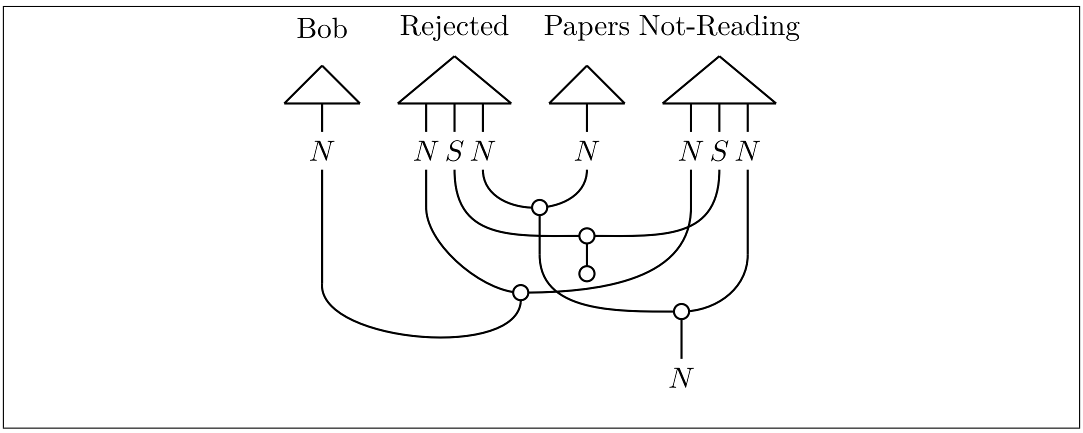


Figure 12: Semantic information flow for the double parasitic gap (normal form).

reading, and ι the unit of the Frobenius coalgebra, this is

$$\overrightarrow{\text{Papers}} \odot (\iota_S \otimes id_N)(\overrightarrow{\text{Bob}}^T \times (\overline{\overline{\text{Rejected}}} \odot \overline{\overline{\text{Not-Reading}}}))$$

The closed linear algebraic form says that we take the elementwise multiplication of both cubes, and contract them with the subject *Bob*; then, we collapse the resulting matrix into a vector and compute the elementwise multiplication of this vector with the vector interpreting the head noun *Papers*.

For the co-argument case of parasitic gapping, we insert the derived Frobenius diagrams for *that* and *whom*, to obtain the initial diagrams of Figures 13 (1e) and 14 (1f), which normalise to the diagrams in Figures 15,16. Note that the lexical specification of *made* and *persuade* is a wrapper around the lexical content of the verbs; since *public* and the phrase *to vote for* are interpreted as $N \otimes S$, their understood subject needs to be supplied, which happens through the use of Frobenius operations in the specification of their consuming verbs. This is the direct analogue of assigning a lambda term $\lambda x.\lambda P.\lambda y.PERSUADE\ x\ (P\ x)\ y$ to *persuade*, where the Frobenius expansion corresponds to variable reuse.

4 Discussion

The concrete modelling presented above produces an interpretation of relative clauses that is analogous to the formal semantics account: seeing elementwise multiplication as an intersective operation (cf. set intersection), the interpretation of *papers that Bob rejected without reading* identifies those papers that were both rejected and not reviewed, by Bob.

In the formal semantics account, the head noun and the relative clause body are both interpreted as functions from individuals to truth values, i.e. characteristic functions of sets of individuals, which allows them to be combined by set intersection. In our vector-based modelling, however, the head noun and the relative clause body are initially sent to different semantic spaces, viz. N for the head noun versus $N \otimes S$ for the relative clause body. This means we need to appeal to the ι operation to effectuate the rank reduction from $N \otimes S$ to N that reduces the interpretation of the relative clause body to a vector that can then be conjoined with the meaning of the head noun. The rank reduction performed by the ι transformation is not a lossless transformation, and it is debatable whether it correctly captures the semantic action we want to associate with the relative pronoun.

As a first step towards a more general model, we abstract away from the specific modelling of the relative pronoun by means of the ι map.

As shown in Figure 17, our type translation for the relative pronoun effectively interprets it as a map from a verb phrase ($N \otimes S$) meaning into an adjectival meaning modifying a (common) noun ($N \otimes N$).

With this generalization, we are not bound anymore to a specific implementation of the relative pronoun meaning, although the proposed account for now gives a workable solution for experimentation.

We suggest here, that a data-driven approach may lend itself for modelling the relative pronoun, as it essentially binds a verb phrase to its adjectival form. For example, a verb phrase can occur in adjectival form, e.g. “papers that were rejected" vs “rejected papers”. In such cases, we would expect to get the same meaning representation, which crucially relies on being able to project either an adjective onto a verb phrase or vice versa. Formulating this as a machine learning problem, is work in progress.

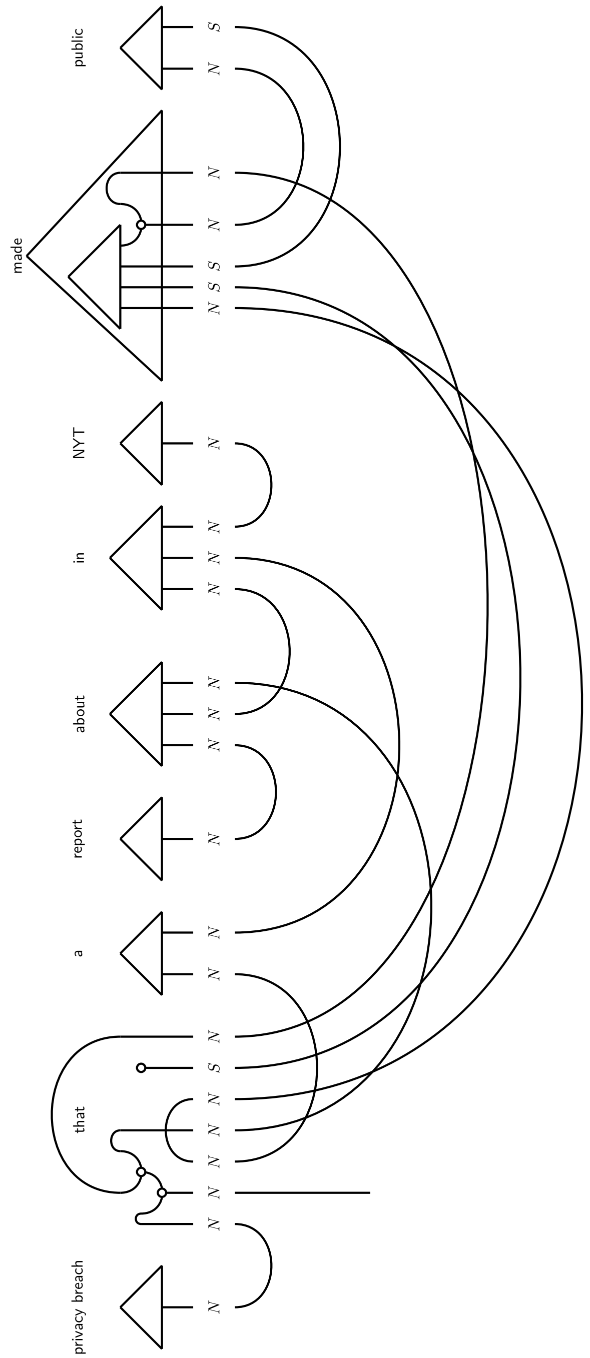


Figure 13: Semantic information flow for the co-argument parasitic gap (1e, initial form).

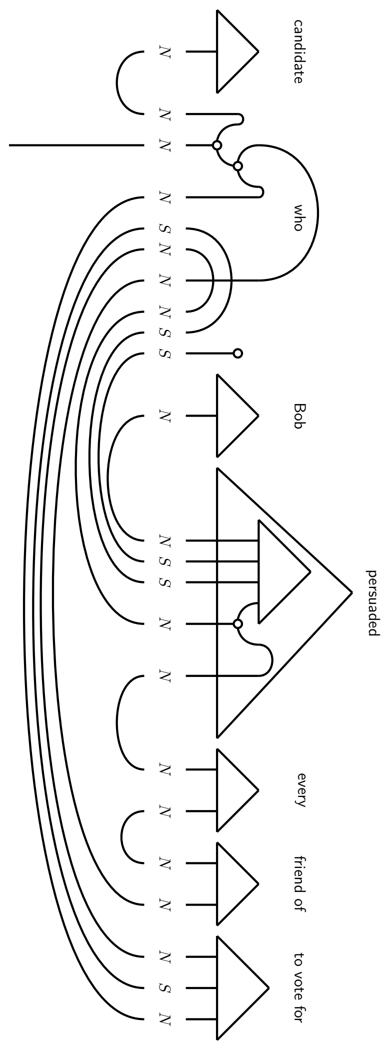


Figure 14: Semantic information flow for the co-argument parasitic gap (1f, initial form).

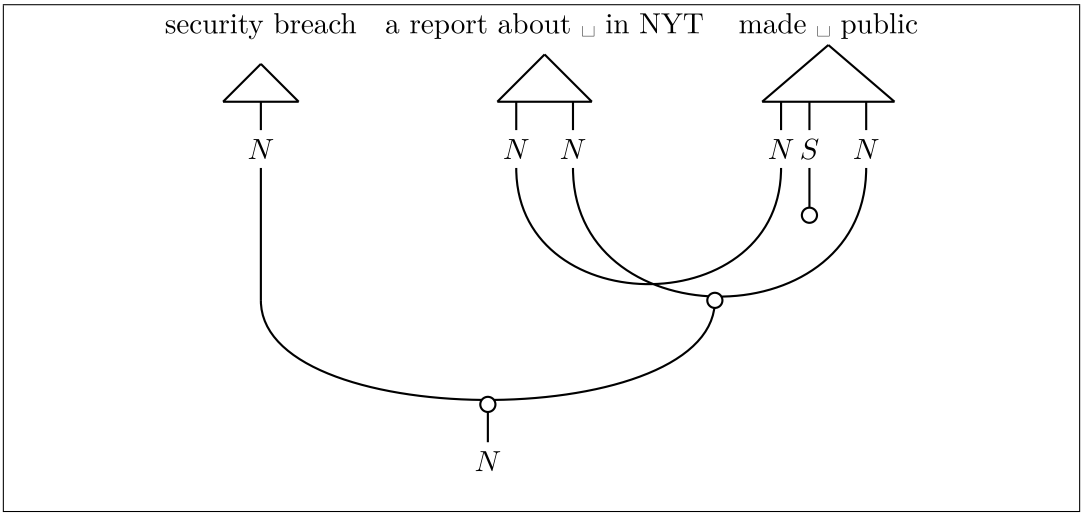


Figure 15: Semantic information flow for the co-argument parasitic gap (1e, normal form).

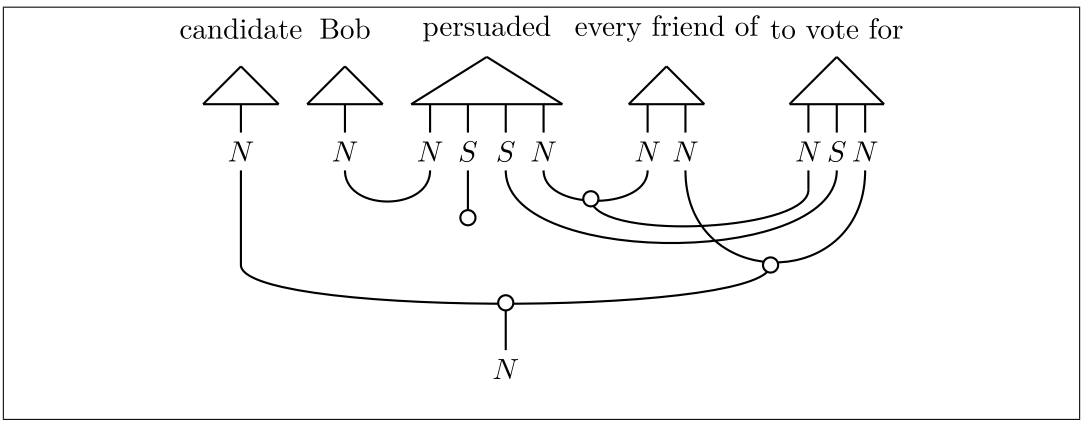


Figure 16: Semantic information flow for the co-argument parasitic gap (1f, normal form).

5 Conclusion/Future Work

We presented a typelogical ditributional account of parasitic gapping, one of the many linguistic phenomena in which some semantic elements are not present in the sentence (or more generally discourse) and therefore their corresponding information needs to be provided from some other syntactic element. Rather than relying on some form of copying and/or movement on the syntax side to provide this in-

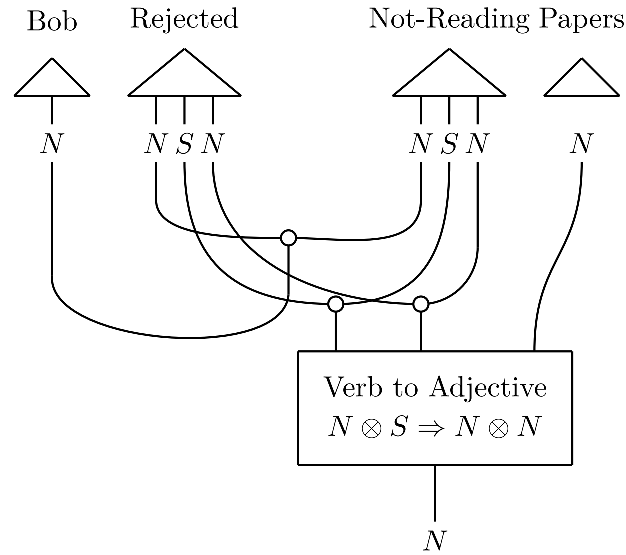


Figure 17: General normal form for a sentence with a parasitic gap; the relative pronoun is now a general map that transforms a verb phrase ($N \otimes S$) into an adjective ($N \otimes N$).

formation (as is the approach for ellipsis with anaphora in [20, 22]), we have solved the problem by using polymorphic typing for function words that play a key role in parasitic gapping (here, *that*, *whom* and *without*).

The polymorphism carries over to the semantics, where we have used Frobenius algebras to interpret them. This enabled us to handle the coordination of multiple gaps, and where the relative pronoun *that* handles the coordination of the head noun with the body of the relative clause and the pronoun *without* coordinates the second gap that exists in the body and which refers to the same head noun. The lexical specifications we use are analogous to a formal semantic modelling, but moreover allow for a more flexible way of representing meaning that may be obtained from data. That resolving gaps is useful in verb disambiguation and sentence similarity tasks has been recently shown [21]. On this point, we discussed a more general normal form in which the behaviour of the relative pronoun is kept abstract. Investigating alternatives to the current modelling with the ι map, and looking into data-driven modelling of the relative pronoun, constitutes work in progress.

References

[1] Coecke, B., Pavlovic, D., Vicary, J.: A new description of orthogonal bases. Mathematical Structures in Computer Science 1, 269–272 (2008)

[2] Culicover, P., Postal, P. (eds.): Parasitic Gaps. MIT Press (2001)

[3] Došen, K.: A brief survey of frames for the Lambek calculus. Mathematical Logic Quarterly 38(1), 179–187 (1992)

[4] Kanovich, M.I., Kuznetsov, S., Scedrov, A.: Undecidability of a newly proposed calculus for CatLog3. In: Bernardi, R., Kobele, G., Pogodalla, S. (eds.) Formal Grammar - 24th International Conference, FG 2019, Riga, Latvia, August 11, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11668, pp. 67–83. Springer (2019)

[5] Kartsaklis, D.: Coordination in categorical compositional distributional semantics. In: Proceedings of the 2016 Workshop on Semantic Spaces at the Intersection of NLP, Physics and Cognitive Science, SLPCS@QPL 2016, Glasgow, Scotland, 11th June 2016. pp. 29–38 (2016)

[6] Lambek, J.: The mathematics of sentence structure. American Mathematical Monthly 65, 154–170 (1958)

[7] Lambek, J.: On the calculus of syntactic types. In: Jakobson, R. (ed.) Structure of Language and its Mathematical Aspects, Proceedings of Symposia in Applied Mathematics, vol. XII, pp. 166–178. American Mathematical Society (1961)

[8] Lambek, J.: Type grammar revisited. In: Lecomte, A., Lamarche, F., Perrier, G. (eds.) Logical Aspects of Computational Linguistics, Second International Conference, LACL '97, Nancy, France, September 22-24, 1997, Selected Papers. Lecture Notes in Computer Science, vol. 1582, pp. 1–27. Springer (1997)

[9] Moortgat, M., Wijnholds, G.: Lexical and derivational meaning in vector-based models of relativisation. In: Proceedings of the 21st Amsterdam Colloquium. p. 55. ILLC, University of Amsterdam (2017), `https://arxiv.org/abs/1711.11513`

[10] Moortgat, M.: Multimodal linguistic inference. Journal of Logic, Language and Information 5(3-4), 349–385 (1996)

[11] Morrill, G.: Type Logical Grammar: Categorial Logic of Signs. Kluwer Academic Publishers, Dordrecht (1994)

[12] Morrill, G.: Parsing/theorem-proving for logical grammar CatLog3. Journal of Logic, Language and Information 28(2), 183–216 (2019)

[13] Morrill, G., Valentín, O.: Computational coverage of TLG: Nonlinearity. In: Proceedings of Third Workshop on Natural Language and Computer Science. vol. 32, pp. 51–63. EasyChair Publications (2015)

[14] Morrill, G., Valentín, O.: On the logic of expansion in natural language. In: Logical Aspects of Computational Linguistics. pp. 228–246. Springer (2016)

[15] Partee, B., Root, M.: Generalized conjunction and type ambiguity. In: Bäuerle, R., Schwarze, C., von Stechow, A. (eds.) Meaning, Use, and Interpretation of Language, pp. 361–383. Walter de Gruyter, Berlin (1983)

[16] Sadrzadeh, M., Clark, S., Coecke, B.: The Frobenius anatomy of word meanings I: subject and object relative pronouns. Journal of Logic and Computation 23(6), 1293–1317 (2013)

[17] Sadrzadeh, M., Clark, S., Coecke, B.: Frobenius anatomy of word meanings 2: possessive relative pronouns. Journal of Logic and Computation 26, 785–815 (2014)

[18] Selinger, P.: A survey of graphical languages for monoidal categories. In: Coecke, B. (ed.) New Structures for Physics, Lecture Notes in Physics, vol. 813, pp. 289–355. Springer, Berlin, Heidelberg (2011)

[19] Steedman, M.: Combinatory grammars and parasitic gaps. Natural Language and Linguistic Theory 5, 403–439 (1987)

[20] Wijnholds, G., Sadrzadeh, M.: Classical copying versus quantum entanglement in natural language: The case of VP-ellipsis. EPTCS 283, 2018, pp. 103-119 pp. 103–119 (2018)

[21] Wijnholds, G., Sadrzadeh, M.: Evaluating composition models for verb phrase elliptical sentence embeddings. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). Association for Computational Linguistics (2019)

[22] Wijnholds, G., Sadrzadeh, M.: A type-driven vector semantics for ellipsis with anaphora using lambek calculus with limited contraction. Journal of Logic, Language and Information (2019)

[23] Wijnholds, G.J.: Coherent diagrammatic reasoning in compositional distributional semantics. In: International Workshop on Logic, Language, Information, and Computation. pp. 371–386. Springer (2017)

 Received 1 July 2020