



From Logical to Distributional Models

Anne Preller

► **To cite this version:**

Anne Preller. From Logical to Distributional Models. Electronic Proceedings in Theoretical Computer Science, EPTCS, 2014, 171, pp.113-131. <10.4204/EPTCS.171.11>. <lirmm-00992378>

HAL Id: lirmm-00992378

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00992378>

Submitted on 17 May 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

From Logical to Distributional Models

Anne Preller

LIRMM,
Montpellier, France

*

preller@lirmm.fr

The paper relates two variants of semantic models for natural language, logical functional models and compositional distributional vector space models, by transferring the logic and reasoning from the logical to the distributional models. The geometrical operations of quantum logic are reformulated as algebraic operations on vectors. A map from functional models to vector space models makes it possible to compare the meaning of sentences word by word.

Keywords: compositional semantics for natural language, compact closed categories, quantum logic, logical models, vector space logic, algebraic connectives

1 Introduction

Semantic models for natural language vary from logical models, e.g. first order models or Montague models, to conceptual models. Conceptual models englobe variants of higher order type theory (1) and the geometrical vector space models based on quantum logic (14), (15), (13). They all involve reasoning, an essential ingredient of compositional semantics, (6).

This paper traces the switch from logical functional models of pregroup grammars to the distributional vector space models of (2) and shows how the predicate logic of functional models changes to the quantum logic of vector space models. It also proposes a way to fill a gap left in (4), namely logic.

Pregroup grammars (7) provide the common mathematical background of compact closed categories, which facilitates the passage from the functional logical models to vector space models. Both kinds of models are implemented by structure preserving functors defined on the *lexical category* $\mathcal{L}(\mathcal{B})$, the free compact closed category generated by a partially ordered set of basic types \mathcal{B} and the entries in a pregroup dictionary $\mathcal{D}_{\mathcal{B}}$. In this study, all functors map entries in the dictionary to vectors in the category $\mathbf{FVect}_{\mathbb{R}}$ of finite dimensional vector spaces over the real numbers.

Structure preserving functors are necessarily compositional, because all strings recognised by a pregroup grammar are morphisms of the lexical category. Indeed, any grammatical analysis produced by a pregroup grammar corresponds to a morphism r of the lexical category. The meaning of the string is defined in the lexical category as the composite of r with the juxtaposition (tensor product) of the lexical entries (vectors). A model implemented by a functor from $\mathcal{L}(\mathcal{B})$ to some compact closed category commutes with the structural operations. Hence the value of the string is computed from the values of the words using the operations of the compact closed category.

The main difference between the functors standing for logical functional models and those standing for vector space models lies in the interpretation of the sentence type. The former models map it to a two-dimensional space of truth-values and the latter to a higher dimensional space of concepts represented by words.

*The author wishes to thank the LIRMM and in particular the research group TEXTE for support

A vector space model interprets the so-called *property words* or *concepts*, i.e. nouns, adjectives and verbs, by vectors in a fixed finite dimensional space W , the basis vectors of which are identified with a set of basic concepts. Words with a logical content like determiners, relative pronouns, connectives (*and*, *not* etc.) and auxiliary verbs are not interpreted. They are ‘noise’. The compositional extension to meanings of strings is given in (4) ”via a strongly monoidal functor from the free pregroup¹ of basic types to the category of finite dimensional vector spaces $\mathbf{FVect}_{\mathbb{R}}$ ” using the Frobenius multiplication in form of the pointwise product \odot for composition.

Logical functional models interpret all words in the lexicon. They map nouns to vectors with coefficients equal to 0 or 1, attributive adjectives to projectors (i.e. linear maps that map any basis vector to itself or to 0) and verbs and predicative adjectives to predicates, (i.e. linear maps that take their values in a space of *truth values*). They interpret logical words as distinguished linear maps that operate on predicates by composition. For instance, if $\text{not} : S \rightarrow S$ is the linear map interpreting the negation *not* and blue the predicate corresponding to the adjective *blue* then $\text{not} \circ \text{blue}$ is the meaning of *not blue*. The induced logic is four-valued and strictly extends first order predicate logic. Besides ‘true’ and ‘false’ it has truth-values ‘meaningless’, e.g. *rocks sleep*, and ‘mixed’, e.g. *rocks are grey*.

The question addressed here is if and how the compositional approach to vector space models can be extended to noise words. Indeed, the compositional approach via a functor does not extend to most logic words. For instance, there is no vector $\overrightarrow{\text{not}}$ for which $\overrightarrow{\text{not}} \odot \overrightarrow{\text{blue}}$ would be orthogonal to $\overrightarrow{\text{blue}}$.

The geometrical logical connectives *and*, *not or*, *if-then* of quantum logic operate on projectors, not on vectors. After reformulating the geometrical operations as algebraic operations, this study shows that the algebraic operations and the corresponding consequence relation define a conditional logic on the concept space similar to the conditional logic of information retrieval in (13). Negation, for instance, coincides with orthogonality.

How adequate is the extension of the vector space model with respect to the meaning of sentences? To test the adequateness, I pair any logical functional model \mathcal{F} with a vector space model \mathcal{M}_C interpreting words in a space C and define a map \mathcal{J} from ‘properties’, e.g. vectors, projectors and predicates, to vectors of C such that \mathcal{M}_C coincides on words with the composite $\mathcal{J} \circ \mathcal{F}$. The model \mathcal{M}_C depends on the model \mathcal{F} , a possible world to which the statements refer, and the choice of basic vectors of C , expressing semantic relations between words, and the grammatical role of words in strings of words. The coefficients of the vectors in \mathcal{M}_C have a truth probabilistic content related to and motivated by quantum logic.

The logical functional model and the vector space model are both functors defined on the lexical category. The value in the functional model is the composite of the word values, e.g. $\mathcal{F}(\text{rocks are grey}) = \mathcal{F}(\text{are}) \circ \mathcal{F}(\text{grey}) \circ \mathcal{F}(\text{rocks})$, whereas the vector space model \mathcal{M}_C uses the pointwise multiplication, e.g. $\mathcal{M}_C(\text{rocks are grey}) = \mathcal{M}_C(\text{rocks}) \odot \mathcal{M}_C(\text{are}) \odot \mathcal{M}_C(\text{grey})$. The meanings of strings in the two models can be compared word by word and operator by operator. Under sufficient conditions, the truth-probabilistic content of words is preserved by strings, i.e. $\mathcal{J}(\mathcal{F}(\text{word}_1 \dots \text{word}_n)) = \mathcal{M}(\text{word}_1 \dots \text{word}_n)$.

In fact, the map \mathcal{J} preserves truth and reflects the algebraic consequence relation. Moreover, \mathcal{J} preserves negation. Under sufficient conditions \mathcal{J} also preserves the binary connectives. In particular, under these conditions the pointwise product commutes with the probability interpretation, because the conjunction of vectors is their pointwise product. The algebraic operations also coincide with the connectives of predicate logic in the degenerate case where individuals play the role of ‘basic concepts’.

¹The authors seem to use the term ‘free pregroup’ in a sense that is not compatible with the existence of a strongly monoidal functor, see Section 3.

2 Lexical semantics of pregroup grammars

The free pregroup $\mathcal{P}(\mathcal{B})$ generated by a partially ordered set \mathcal{B} introduced in (7) is a partially preordered monoid in which every element a has a right adjoint a^r and a left adjoint a^ℓ characterised by the equalities

$$aa^r \leq 1 \leq a^r a \quad a^\ell a \leq 1 \leq aa^\ell.$$

Semantics requires a slightly modified definition, namely the free compact closed category $\mathcal{C}(\mathcal{B})$ generated by \mathcal{B} , introduced in (11). The only difference between the two versions is that $\mathcal{P}(\mathcal{B})$ identifies all morphisms of $\mathcal{C}(\mathcal{B})$ that have a common domain and a common codomain. This difference is essential when compositional semantics in vector spaces are mediated by a functor. An example of morphisms identified in $\mathcal{P}(\mathcal{B})$ leading to different meanings is given in (11). See also Fact 1 in Section 3 showing that a strongly monoidal functor from the free pregroup maps every element of the free pregroup to a space of dimension at most one.

Recall that a *monoidal category* consists of a category \mathcal{C} , a bifunctor, (denoted somewhat misleadingly by the tensor symbol) \otimes , a distinguished object I , the *unit of the bifunctor*, and natural isomorphisms $\alpha_{ABC} : (A \otimes B) \otimes C \rightarrow A \otimes (B \otimes C)$, $\lambda_A : A \rightarrow I \otimes A$ and $\rho_A : A \rightarrow A \otimes I$ subject to the coherence conditions of (9). A monoidal category is *symmetric* if there is a natural isomorphism $\sigma_{AB} : A \otimes B \rightarrow B \otimes A$ such that $\sigma_{AB}^{-1} = \sigma_{BA}$, again subject to the coherence conditions of (9). A monoidal category is *compact closed* if for every object A there are objects A^r and A^ℓ , called *right adjoint* and *left adjoint* of A respectively, and morphisms $\eta_A : I \rightarrow A^r \otimes A$, $\varepsilon_A : A \otimes A^r \rightarrow I$, $\eta_{A^\ell} : I \rightarrow A \otimes A^\ell$, $\varepsilon_{A^\ell} : A^\ell \otimes A \rightarrow I$ satisfying

$$\lambda_A^{-1} \circ (\varepsilon_A \otimes 1_A) \circ \alpha_{AA^r A}^{-1} \circ (1_A \otimes \eta_A) \circ \rho_A = 1_A, \quad \lambda_{A^\ell}^{-1} \circ (\varepsilon_{A^\ell} \otimes 1_{A^\ell}) \circ \alpha_{A^\ell A A^\ell}^{-1} \circ (1_{A^\ell} \otimes \eta_{A^\ell}) \circ \rho_{A^\ell} = 1_{A^\ell}.$$

For notational convenience, the associativity isomorphisms α_{ABC} and the unit isomorphisms λ_A and ρ_A are replaced by identities, e.g. $(A \otimes B) \otimes A = A \otimes (B \otimes C)$, $A = I \otimes A$ and $A \otimes I$. The equalities above become the *adjoint equalities*

$$(\varepsilon_A \otimes 1_A) \circ (1_A \otimes \eta_A) = 1_A, \quad (\varepsilon_{A^\ell} \otimes 1_{A^\ell}) \circ (1_{A^\ell} \otimes \eta_{A^\ell}) = 1_{A^\ell}. \quad (1)$$

A functor between compact closed categories *preserves the compact closed structure* if it commutes with the tensor product, the tensor unit and right and left adjoints up to natural isomorphisms.

Every morphism of the free compact closed category generated by some category \mathcal{C} can be designated by a ‘normal’ graph where all links are labelled by morphisms of \mathcal{C} and all paths have length 1. In the case where the label is an identity it may be omitted. The graph displays the domain above, the codomain below. Vertical links correspond to right or left adjoints of morphisms of \mathcal{C} , overlinks to names and the underlinks to conames of morphisms of \mathcal{C} . For example, the right and the left adjoint of a basic morphism $f : \mathbf{a} \rightarrow \mathbf{b}$ are represented by the graphs

$$f^r : \mathbf{b}^r \rightarrow \mathbf{a}^r = \begin{array}{c} \mathbf{b}^r \\ \uparrow f \\ \mathbf{a}^r \end{array} \quad f^\ell : \mathbf{b}^\ell \rightarrow \mathbf{a}^\ell = \begin{array}{c} \mathbf{b}^\ell \\ \uparrow f \\ \mathbf{a}^\ell \end{array}$$

Any morphism $f : a \rightarrow b$ has two names

$$\begin{array}{ccc}
 \text{right name} & & \text{left name} \\
 \eta_f & = (f^r \otimes 1_b) \circ \eta_b & \eta_{f^\ell} = (f \otimes 1_{a^\ell}) \circ \eta_{a^\ell} \\
 I & & I \\
 \begin{array}{c} \xrightarrow{f} \\ a^r \otimes b \end{array} & = & \begin{array}{c} \xrightarrow{f^r} \\ b^r \otimes b \\ \uparrow f \\ a^r \otimes b \end{array} \\
 \end{array}
 \qquad
 \begin{array}{ccc}
 \begin{array}{c} \xrightarrow{f} \\ b \otimes a^\ell \end{array} & = & \begin{array}{c} \xrightarrow{f} \\ a \otimes a^\ell \\ \downarrow f \\ b \otimes a^\ell \end{array}
 \end{array}$$

Analogous definitions and notations apply to conames, namely ε_f and ε_{f^ℓ}

$$\begin{array}{ccc}
 \text{right coname} & & \text{left coname} \\
 \varepsilon_f & = \varepsilon_b \circ (f \otimes 1_{b^r}) & \varepsilon_{f^\ell} = \varepsilon_{b^\ell} \circ (1_{b^\ell} \otimes f) \\
 \begin{array}{c} \xrightarrow{f} \\ a^r \otimes b \end{array} & = & \begin{array}{c} \xrightarrow{f} \\ a \otimes b^r \\ \downarrow f \\ b \otimes b^r \end{array} \\
 I & & I \\
 \begin{array}{c} \xrightarrow{f} \\ b^\ell \otimes a \end{array} & = & \begin{array}{c} \xrightarrow{f} \\ b^\ell \otimes a \\ \downarrow f \\ b^\ell \otimes b \end{array} \\
 I & & I
 \end{array}$$

The equality of the composite graph on the right to the normal graph on the left is a particular instance of the so-called ‘yanking’. Start at the tail of any link situated in the top or bottom line and follow the oriented links until a head situated in the top or bottom line is reached. Replace the whole path by a single link, labelled by the composite of the labels in the order they are encountered. In the case of names, yanking works by definition. The general case concerning the composite of two arbitrary graphs follows from the adjoint axioms (1).

$$(\varepsilon_a \otimes 1_a) \circ (1_a \otimes \eta_a) = \begin{array}{c} a \\ \swarrow \quad \searrow \\ a \otimes a^r \otimes a \\ \swarrow \quad \searrow \\ a \end{array} = \begin{array}{c} a \\ \downarrow \\ a \end{array} = \begin{array}{c} a \\ \swarrow \quad \searrow \\ a \otimes a^\ell \otimes a \\ \swarrow \quad \searrow \\ a \end{array} = (\varepsilon_{a^\ell} \otimes 1_{a^\ell}) \circ (1_{a^\ell} \otimes \eta_{a^\ell})$$

The same morphisms composed in the opposite order result in the normal graph on the right

$$(1_a \otimes \eta_a) \circ (\varepsilon_a \otimes 1_a) = \begin{array}{c} a \otimes a^r \otimes a \\ \swarrow \quad \searrow \\ a \\ \swarrow \quad \searrow \\ a \otimes a^r \otimes a \end{array} = \begin{array}{c} a \otimes a^r \otimes a \\ \swarrow \quad \searrow \\ a \\ \swarrow \quad \searrow \\ a \otimes a^r \otimes a \end{array}$$

The free compact closed category generated by an arbitrary category has the ‘normal form property’, namely there is a one-to-one correspondence between morphisms and graphs where all paths are reduced

to a single link. This implies that

$$1_{\mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a}} = \begin{array}{c} \mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a} \\ \downarrow \quad \uparrow \quad \downarrow \\ \mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a} \end{array} \neq \begin{array}{c} \mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a} \\ \swarrow \quad \searrow \\ \mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a} \end{array} = (1_{\mathbf{a}} \otimes \eta_{\mathbf{a}}) \circ (\varepsilon_{\mathbf{a}} \otimes 1_{\mathbf{a}}).$$

The objects of the free compact closed category $\mathcal{C}(\mathcal{B})$ generated by a partially ordered set \mathcal{B} are called *types*, among them are the elements of \mathcal{B} , called *basic types*. A *simple type* has the form $\dots, \mathbf{a}^{(-2)} = \mathbf{a}^{\ell\ell}, \mathbf{a}^{(-1)} = \mathbf{a}^{\ell}, \mathbf{a}^{(0)} = \mathbf{a}, \mathbf{a}^{(1)} = \mathbf{a}^r, \mathbf{a}^{(2)} = \mathbf{a}^{rr}, \dots$ where the \mathbf{a} is a basic type. Any object of $\mathcal{C}(\mathcal{B})$ can be written as a finite string of simple types where juxtaposition plays the role of the monoidal bifunctor. Clearly, $\mathcal{C}(\mathcal{B})$ is not symmetric

A pregroup lexicon consists of pairs $\text{word} : T$ where word is a word of natural language and T a type. One can view each entry $\text{word} : T$ as a formal expression $\overline{\text{word}} : I \rightarrow T$ in the language of compact closed categories. For example,

$\text{no} : \mathbf{s} \otimes \mathbf{s}^{\ell} \otimes \mathbf{n}_2 \otimes \mathbf{c}_2^{\ell}$	$\overline{\text{no}} : I \rightarrow \mathbf{s} \otimes \mathbf{s}^{\ell} \otimes \mathbf{n}_2 \otimes \mathbf{c}_2^{\ell}$
$\text{new} : \mathbf{n}_2 \otimes \mathbf{c}_2^{\ell}$	$\overline{\text{new}} : I \rightarrow \mathbf{c}_2 \otimes \mathbf{c}_2^{\ell}$
$\text{triangles} : \mathbf{c}_2$	$\overline{\text{triangles}} : I \rightarrow \mathbf{c}_2$
$\text{are} : \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n}_2$	$\overline{\text{are}} : I \rightarrow \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n}_2$
$\text{blue} : \mathbf{n}^r \otimes \mathbf{p}$	$\overline{\text{blue}} : I \rightarrow \mathbf{n}^r \otimes \mathbf{p}$

Here $\mathbf{c}_2 \leq \mathbf{n}_2 \leq \mathbf{n}$ are the basic types standing for plural common nouns, plural noun phrases and noun phrases where the number does not matter, in that order. The basic types $\mathbf{p} \leq \mathbf{s}$ correspond to predicative adjectives and to sentences. The reader can find a more comprehensive grammar of English in (8).

Every lexical entry $\overline{\text{word}} : I \rightarrow T = \mathbf{a}_1^{(z_1)} \otimes \dots \otimes \mathbf{a}_n^{(z_n)}$ creates a ‘lexical morphism’ $\text{word} : \mathbf{a}_{i_1} \otimes \dots \otimes \mathbf{a}_{i_m} \rightarrow \mathbf{a}_j$ where z_{i_k} is odd for $k = 1, \dots, m$ and z_j is even. For example,

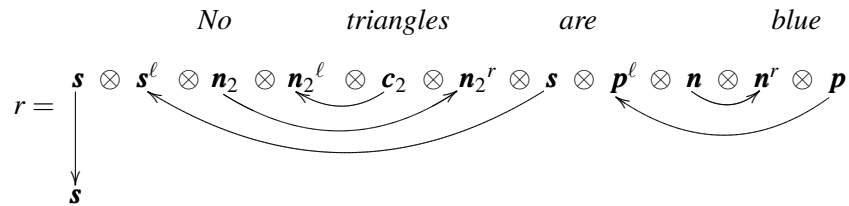
$$\begin{array}{c} \overline{\text{blue}} = \begin{array}{c} I \\ \swarrow \quad \searrow \\ \mathbf{n}^r \otimes \mathbf{p} \end{array} \text{blue}, \quad \begin{array}{c} \mathbf{n} \\ \downarrow \text{blue} \\ \mathbf{p} \end{array} \\ \overline{\text{new}} = \begin{array}{c} I \\ \swarrow \quad \searrow \\ \mathbf{c}_2 \otimes \mathbf{c}_2^{\ell} \end{array} \text{new}, \quad \begin{array}{c} \mathbf{c}_2 \\ \downarrow \text{new} \\ \mathbf{c}_2 \end{array} \\ \overline{\text{triangles}} = \begin{array}{c} I \\ \downarrow \text{triangles} \\ \mathbf{c}_2 \end{array} \\ \overline{\text{are}} = (1_{\mathbf{n}_2^r} \otimes \eta_{\text{are}^{\ell}} \otimes 1_{\mathbf{n}}) \circ \eta_{\mathbf{n}_2} = \begin{array}{c} I \\ \swarrow \quad \searrow \\ \mathbf{n}_2^r \otimes \mathbf{n} \\ \swarrow \quad \searrow \\ \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n} \\ \swarrow \quad \searrow \\ \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n} \end{array} = \begin{array}{c} I \\ \swarrow \quad \searrow \\ \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n} \\ \swarrow \quad \searrow \\ \mathbf{n}_2^r \otimes \mathbf{s} \otimes \mathbf{p}^{\ell} \otimes \mathbf{n} \end{array}, \quad \begin{array}{c} \mathbf{p} \\ \downarrow \text{are} \\ \mathbf{s} \end{array} \\ \overline{\text{no}} = \eta_{\text{not}^{\ell}} \otimes \eta_{\mathbf{n}_2^{\ell}} = \begin{array}{c} I \\ \swarrow \quad \searrow \\ \mathbf{s} \otimes \mathbf{s}^{\ell} \otimes \mathbf{n}_2 \otimes \mathbf{n}_2^{\ell} \\ \swarrow \quad \searrow \\ \mathbf{s} \otimes \mathbf{s}^{\ell} \otimes \mathbf{n}_2 \otimes \mathbf{n}_2^{\ell} \end{array}, \quad \begin{array}{c} \mathbf{s} \\ \downarrow \text{no} \\ \mathbf{s} \end{array} \end{array}$$

Labelled links correspond to lexical morphisms like $\text{not} : \mathbf{s} \rightarrow \mathbf{s}$, $\text{blue} : \mathbf{n} \rightarrow \mathbf{p}$ and $\text{new} : \mathbf{c}_2 \rightarrow \mathbf{n}_2$. Unlabelled links correspond to (in)equalities of basic types $\text{in}_{ab} : \mathbf{a} \rightarrow \mathbf{b}$.

Thus, every pregroup lexicon determines a *lexical category* $\mathcal{L}(\mathcal{B})$, namely the free compact closed category generated by the partially ordered set of basic types and the lexical morphisms. A *basic morphism* is an (in)equality $in_{ab} : \mathbf{a} \rightarrow \mathbf{b}$ between elements of \mathcal{B} or a lexical morphism $word : \mathbf{a}_1 \otimes \dots \otimes \mathbf{a}_m \rightarrow \mathbf{b}$. Note that the set of basic morphisms does not form a category. The generating category of $\mathcal{L}(\mathcal{B})$ is the monoidal category generated by \mathcal{B} and the lexical morphisms. The lexical category is the mathematical tool of a pregroup grammar that provides both grammatical analysis and meanings of words and strings of words.

Call *reduction* any morphism $r : T_1 \dots T_n \rightarrow \mathbf{b}$ of the free compact closed category involving (in)equalities or counits of (in)equalities of basic types only. We omit all labels in the graphical picture, because they are uniquely determined by the tail and the head of the link. The basic type at the tail is necessarily less or equal to the basic type at the head.

A string of words $word_1 \dots word_n$ is *grammatical* if there are entries $word_i : T_i$ in the lexicon, a basic type \mathbf{b} and a reduction $r : T_1 \dots T_n \rightarrow \mathbf{b}$. For instance.

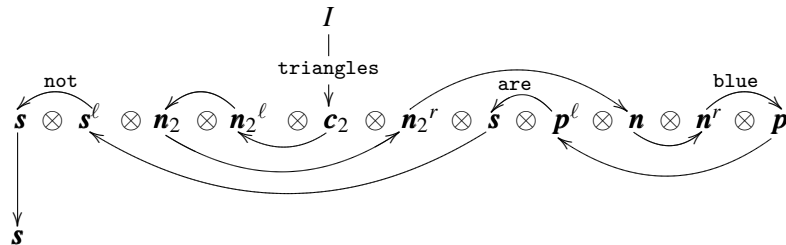


Meanings of grammatical strings are best defined abstractly in the lexical category $\mathcal{L}(\mathcal{B})$. The *meaning* of the string $word_1 \dots word_n$ recognised by the reduction r is

$$r \circ (\overline{word_1} \otimes \dots \otimes \overline{word_n}).$$

For example,

$$r \circ (\overline{\text{no}} \otimes \overline{\text{triangles}} \otimes \overline{\text{are}} \otimes \overline{\text{blue}}) =$$



$$\begin{aligned} &= \text{not} \circ \text{are} \circ \text{blue} \circ in_{n_2 n} \circ in_{c_2 n_2} \circ \text{triangles} \\ &= \text{not} \circ \text{are} \circ \text{blue} \circ in_{c_2 n} \circ \text{triangles}. \end{aligned}$$

Any compact closed structure preserving functor from the lexical category \mathcal{F} into an arbitrary compact closed category \mathcal{C} provides us with a compositional interpretation of grammatical strings. The value of the string is computed from the values of the words with the help of the operations (tensor, composition etc.) of the category. Assume, for instance, that \mathcal{F} is a compact closed structure preserving functor satisfying

$$\begin{aligned} N = \mathcal{F}(\mathbf{n}) = \mathcal{F}(\mathbf{n}_2) = \mathcal{F}(\mathbf{c}_2), & \quad S = \mathcal{F}(\mathbf{s}) = \mathcal{F}(\mathbf{p}), \\ \mathcal{F}(in_{ab}) = 1_{\mathcal{F}(\mathbf{b})}, & \quad \mathcal{F}(\text{are}) = 1_S. \end{aligned}$$

Then

$$\mathcal{F}(r \circ (\overline{\text{no}} \otimes \overline{\text{triangles}} \otimes \overline{\text{are}} \otimes \overline{\text{blue}})) = \mathcal{F}(\text{not}) \circ \mathcal{F}(\text{blue}) \circ \mathcal{F}(\text{triangles}). \quad (2)$$

3 Vector space models as functors

A functor between monoidal categories is *strongly monoidal* if it commutes with the tensor product and the tensor unit up to natural isomorphisms. If both categories are compact closed a strongly monoidal functor also commutes with the units and the counits of adjunction up to natural isomorphisms, hence the functor preserves the compact closed structure.

The vector space models of (4) are "strongly monoidal functors from the free pregroup to the full subcategory \mathbf{FVect}_W of $\mathbf{FVect}_{\mathbb{R}}$ formed by the tensor powers of some chosen space W ". The subcategory \mathbf{FVect}_W is a compact closed subcategory of $\mathbf{FVect}_{\mathbb{R}}$. If W has dimension greater than one, however, such a functor does not exist.

Fact 1. *There is no monoidal structure preserving functor from $\mathcal{P}(\mathcal{B})$ to $\mathbf{FVect}_{\mathbb{R}}$ that maps a basic type to a space of dimension greater than one.*

Proof. Suppose to the contrary that $A = \mathcal{F}(\mathbf{a})$ has dimension at least two and that \mathcal{F} is a functor from $\mathcal{P}(\mathcal{B})$ to $\mathbf{FVect}_{\mathbb{R}}$ that preserves the monoidal structure and, as a consequence, also the compact structure up to natural isomorphisms. Then from $1_{\mathbf{a} \otimes \mathbf{a}^r \otimes \mathbf{a}} = (1_{\mathbf{a}} \otimes \eta_{\mathbf{a}}) \circ (\varepsilon_{\mathbf{a}} \otimes 1_{\mathbf{a}})$ in $\mathcal{P}(\mathcal{B})$ follows that $f = (1_A \otimes \eta_A) \circ (\varepsilon_A \otimes 1_A)$ is an isomorphism in $\mathbf{FVect}_{\mathbb{R}}$. Assume $a_1, a_2 \in A$ are different basis vectors. Then $(\varepsilon_A \otimes 1_A)(a_1 \otimes a_2 \otimes a_1) = 0$, because the counit $\varepsilon_A : A \otimes A \rightarrow I$ is the inner product of the space A . It follows that $f(a_1 \otimes a_2 \otimes a_1) = 0$ and therefore has no inverse. \square

This problem disappears when the free pregroup $\mathcal{P}(\mathcal{B})$ is replaced by the free compact closed category $\mathcal{C}(\mathcal{B})$. A functor defined on $\mathcal{C}(\mathcal{B})$, however, is not enough, because it does not interpret the lexical entries. Adjectives in attributive position, for instance, have type $\mathbf{c}_2 \otimes \mathbf{c}_2^\ell$. The left unit is the only morphism of $\mathcal{C}(\mathcal{B})$ with domain I and codomain $\mathbf{c}_2 \otimes \mathbf{c}_2^\ell$, because the identity is the only endomorphism of the basic type \mathbf{c}_2 in the free category. We must add the lexical morphisms and define a functor on the lexical category $\mathcal{L}(\mathcal{B})$.

Choosing a vector space V and defining the word vectors in V from a corpus is a complex task. Here, we are only interested in the result, namely a map M_V from the entries of a pregroup lexicon to vectors in V , called *vector space model*. The compositional extension of a M_V via a functor such that the meaning of a string is again a vector in V , namely the pointwise product of the word vectors has a straight forward definition, due to the following fact.

Fact 2. *Let V be a commutative monoid with binary operation \star and neutral element e . Then the following definitions define a compact closed category V_\star*

V is the unique object of V_\star

The elements $v \in V$ are the morphisms $v : V \rightarrow V$ of V_\star

$$v_1 \circ v_2 = v_1 \star v_2, \quad 1_V = e$$

$$V \otimes V = V$$

$$v_1 \otimes v_2 = v_1 \star v_2$$

$$V^r = V = V^\ell, \quad \eta_V = e = \varepsilon_V$$

With these definitions, every morphism of V_\star is equal to its name and coname.

Proof. The proof is straight forward. To see that V_\star is a monoidal category with tensor unit V we must show \star is a bifunctor from the product category $V_\star \times V_\star$ into V_\star , that is to say we must show the two equalities

$$1_V \otimes 1_V = 1_{V \otimes V} \quad (v_1 \otimes v_2) \circ (w_1 \otimes w_2) = (v_1 \circ w_1) \otimes (v_2 \circ w_2).$$

The first follows immediately from the definition. The second holds, because the operation \star is commutative. Compact closure is as easily checked. Indeed,

$$\begin{aligned}\eta_V = 1_V : V \rightarrow V = V \otimes V, \quad \varepsilon_V = 1_V : V = V \otimes V \rightarrow V \\ (\varepsilon_V \otimes 1_V) \circ (1_V \otimes \eta_V) = (e \star e) \star (e \star e) = e = 1_V.\end{aligned}$$

The last assertion that $\lceil m \rceil = m = \lfloor m \rfloor$ is a straight forward consequence of the definitions. \square

If \star is a commutative binary operation on a vector space V then any compact closed structure preserving functor from the lexical category to V_\star is a vector-based model in the sense of (10).

There are two obvious choices for the binary operation \star in the case of a finite dimensional vector space V . One is the familiar addition of vectors. The other one is the pointwise product \odot of vectors used in (3). It is defined in terms of any orthonormal basis $A = \{a_1, \dots, a_n\}$ of V by the equality

$$\left(\sum_{i=1}^n \alpha_i a_i\right) \odot \left(\sum_{i=1}^n \beta_i a_i\right) = \sum_{i=1}^n (\alpha_i \beta_i) a_i.$$

The neutral element of \odot is the vector $\vec{1} = \sum_{i=1}^n a_i$. We shall see that the pointwise multiplication also equips the vector space V with a logic that is the vector version of the geometrical logic of projectors.

Any vector space model M_V extends to a unique compact closed structure preserving functor \mathcal{M}_V from the lexical category $\mathcal{L}(\mathcal{B})$ into the compact closed category V_\odot satisfying

$$\begin{aligned}\mathcal{M}(\mathbf{a}) = V \quad \mathcal{M}(in_{\mathbf{ab}}) = \vec{1}, \text{ for all } \mathbf{a}, \mathbf{b} \in \mathcal{B} \\ \mathcal{M}(\text{word}_T) = M(\text{word} : T) \in V \text{ for all lexical entries } \text{word} : T.\end{aligned}$$

Indeed, such a functor exists and is unique because the lexical category is the free compact closed category generated by the lexical words and the inequalities of basic types. The subscript V will be omitted unless this leads to confusion.

Note that the functor \mathcal{M} satisfies $\mathcal{M}(T) = V$ for every type T . Moreover, $\mathcal{M}(r) = \vec{1}$ for any reduction $r : T_1 \otimes \dots \otimes T_n \rightarrow \mathbf{b}$ of the pregroup grammar, because a reduction is an expression of monoidal categories involving only conames of inequalities of basic types. Every morphism of V_\odot is equal to its name(s). Hence, the value assigned to a grammatical string $\text{word}_1 \dots \text{word}_n$ is simply the pointwise product of the vectors

$$\mathcal{M}(r \circ (\overline{\text{word}}_{1T_1} \otimes \dots \otimes \overline{\text{word}}_{nT_n})) = M(\text{word}_1 : T_1) \odot \dots \odot M(\text{word}_n : T_n). \quad (3)$$

The result still depends on the reduction (the pregroup version of syntactical analysis), because it is the reduction that chooses the types T_i . Evidently, this definition of compositional models avoids problem of variable tensor powers.

4 Logical functional models as functors

The logical functional models are extensions of first order models to vector spaces. The resulting logic properly extends first order logic, because not only individuals but also sets of individuals have truth values.

A functor \mathcal{F} from the lexical category to the category of finite-dimensional vector spaces over the field of real numbers is a *logical functional model* if it maps the sentence type \mathbf{s} to a two-dimensional

space $S = \mathcal{F}(\mathbf{s})$ with ‘canonical’ basis vectors \top and \perp , nouns to sums of basic vectors of $N = \mathcal{F}(\mathbf{n})$, determiners and attributive adjectives to projectors, verbs and predicative adjectives to ‘predicates’ and logical words to ‘logical connectives’.

PREDICATES

Let $A = \{a_1, \dots, a_n\}$ be an orthonormal basis of $\mathbb{R}^n = V_A$. A linear map $p : V_A \rightarrow S$ is a *predicate* if $p(a) \in \{0, \top, \perp\}$ for any $a \in A$. It is a *predicate on A* if $p(a) \neq 0$ for all $a \in A$.

Examples are the linear maps $\text{true} : V_A \rightarrow S$ and $\text{false} : V_A \rightarrow S$ satisfying

$$\text{true}(a_i) = \top \text{ respectively } \text{false}(a_i) = \perp \text{ for } i = 1, \dots, n.$$

A linear predicate p ‘counts’ the number of basis vectors for which it takes the value \top . Identify any subset $B = \{a_{i_1}, \dots, a_{i_m}\}$ of m distinct basis vectors with the vector $\vec{B} = \sum_{l=1}^m a_{i_l}$. Let n_{pB} be the number of elements of B for which p returns the value \top . Assume that p is a predicate on A . The following holds

COUNTING PROPERTY

$$p(\vec{B}) = n_{pB}\top + (m - n_{pB})\perp \text{ and } n_{\text{true}B} = |B|. \quad (4)$$

The counting property gives us a clue about how to generalise truth-values to real vector spaces. Recall that a vector v is co-linear to a vector w if there is a scalar $\alpha \neq 0$ such that $v = \alpha w$

TRUTH-VALUES

Let p be a linear predicate on A and X any vector of V_A . We say that

- $p(X)$ is *true* if $p(X)$ is co-linear to \top
- $p(X)$ is *false* if $p(X)$ is co-linear to \perp
- $p(X)$ is *mixed* if $p(X) = \alpha\top + \beta\perp$ for some $\alpha \neq 0, \beta \neq 0$
- $p(X)$ is *mute* if $p(X) = 0$.

The corresponding logic has four truth values, namely ‘true’, ‘false’, ‘mixed’ and ‘mute’. A linear predicate assigns to a basis vector (individual) either ‘true’ or ‘false’ or ‘mute’. If the latter is the case, the predicate has no answer to the question whether the individual has the property or not.

Truth-values are invariant under scaling. The vectors X and λX have identical truth-values for $\lambda \neq 0$. Saying ‘ $p(X)$ is not true’ means that $p(X)$ is not co-linear to the basis vector \top . This does not imply that ‘ p is false on X ’.

LOGICAL CONNECTIVES

The *logical connectives* are the linear maps $\text{not} : S \rightarrow S$, $\text{and} : S \otimes S \rightarrow S$, $\text{or} : S \otimes S \rightarrow S$ and $\text{ifthen} : S \otimes S \rightarrow S$ determined by their values on the basis vectors $z \in \{\top \otimes \top, \top \otimes \perp, \perp \otimes \top, \perp \otimes \perp\}$ thus

$$\begin{aligned} \text{and}(z) &= \begin{cases} \top & \text{if } z = \top \otimes \top \\ \perp & \text{else} \end{cases} & \text{or}(z) &= \begin{cases} \perp & \text{if } z = \perp \otimes \perp \\ \top & \text{else} \end{cases} \\ \text{ifthen}(z) &= \begin{cases} \perp & \text{if } z = \top \otimes \perp \\ \top & \text{else} \end{cases} & \text{not}(\top) &= \perp & \text{not}(\perp) &= \top. \end{aligned}$$

The logical connectives induce a Boolean algebra structure on the set of predicates on A with largest element true . Let $d_A : V_A \rightarrow V_A \otimes V_A$ be the unique linear map satisfying $d_A(a_i) = a_i \otimes a_i$ for $i = 1, \dots, n$ and

$$\langle p, q \rangle = (p \otimes q) \circ d_A : V_A \rightarrow S \otimes S.$$

Then the linear maps

$$\text{not} \circ p : V_A \rightarrow S, \text{ and} \circ \langle p, q \rangle, \text{ or} \circ \langle p, q \rangle, \text{ ifthen} \circ \langle p, q \rangle : V_A \rightarrow S$$

are predicates on A .

LOGICAL CONSEQUENCE RELATION

A predicate q is said to be a *logical consequence* of a predicate p if

$$\text{ifthen} \circ \langle p, q \rangle = \text{true}.$$

The logic introduced above extends first order predicate logic. Indeed, assume that a vector $X = \sum_{i=1}^n \alpha_i a_i \neq 0$ satisfies $\alpha_i \geq 0$, for $i = 1, \dots, n$, and let $\bar{X} = \{a_{i_1}, \dots, a_{i_m}\}$ be the subset of basis vectors a_{i_k} for which $\alpha_{i_k} \neq 0$. Then the following holds

FUNDAMENTAL PROPERTY

$$\begin{aligned} p \text{ is true on } X &\Leftrightarrow \forall x(x \in \bar{X} \Rightarrow p(x) = \top) \\ p \text{ is false on } X &\Leftrightarrow \forall x(x \in \bar{X} \Rightarrow p(x) = \perp) \\ p \text{ is mixed on } X &\Leftrightarrow \exists x \exists y(x, y \in \bar{X} \ \& \ p(x) = \top \ \& \ p(y) = \perp). \end{aligned} \quad (5)$$

Words are interpreted by vectors with non-negative coordinates in the functional vector models. Hence, the Fundamental Property applies to all of them.

The linear map not plays a double role in this logic. It is negation when the predicate is applied to a basis vector. For instance, *joe is tall* versus *joe is not tall*. It is the opposite, when applied to an arbitrary vector, a second order entity. For instance, *all boys are tall* versus *no boys are tall*. The latter assertion implies the negation of the former, but the converse does not hold.

Example 1. *Property? : yes/no*

Consider a game involving chips that come in different shapes and colours. Each shape is coloured with one or several of the colours *red*, *yellow* and *blue*. The machine that distributes the chips can recognise colours, but not shapes. Players who want a certain shape therefore must describe the shape, say *triangle*, *square*, *circle*, in terms of colour combinations.

A player who believes in functional models observes thirty chips $A = \{a_1, \dots, a_{30}\}$ extracted from the machine and represents them by a functional model \mathcal{F} , namely

$$\begin{aligned} \mathcal{F}(\mathbf{n}) &= N = V_A \\ \mathcal{F}(\text{triangle}) &= a_1 + \dots + a_{10} & \mathcal{F}(\text{square}) &= a_{11} + \dots + a_{20} & \mathcal{F}(\text{circle}) &= a_{21} + \dots + a_{30} \\ \mathcal{F}(\text{new})(a_i) &= a_i \text{ if } i = 5, 7 - 15, 20, 25, 30 & \mathcal{F}(\text{new})(a_i) &= 0 \text{ else} \\ \mathcal{F}(\text{blue})(a_i) &= \top \text{ if } i = 16 - 20 & \mathcal{F}(\text{blue})(a_i) &= \perp \text{ else} \\ \mathcal{F}(\text{red})(a_i) &= \top \text{ if } i = 1 - 9, 11 - 20 & \mathcal{F}(\text{red})(a_i) &= \perp \text{ else} \\ \mathcal{F}(\text{yellow})(a_i) &= \top \text{ if } i = 7 - 15, 21 & \mathcal{F}(\text{yellow})(a_i) &= \perp \text{ else.} \end{aligned}$$

Because of his preference for new chips, he computes the noun phrases *new triangles*, *new squares* etc.

$$\begin{aligned} \mathcal{F}(\text{new} \circ \text{square}) &= \mathcal{F}(\text{new}) \circ \mathcal{F}(\text{square}) = \mathcal{F}(\text{new})(a_{11} + \dots + a_{20}) \\ &= a_{11} + a_{12} + a_{13} + a_{14} + a_{15} + a_{20} \\ \mathcal{F}(\text{new} \circ \text{triangle}) &= \mathcal{F}(\text{new}) \circ \mathcal{F}(\text{triangle}) = \mathcal{F}(\text{new})(a_1 + \dots + a_{10}) \\ &= a_5 + a_7 + a_8 + a_9 + a_{10} \\ \mathcal{F}(\text{new} \circ \text{circle}) &= \mathcal{F}(\text{new}) \circ \mathcal{F}(\text{circle}) = \mathcal{F}(\text{new})(a_{21} + \dots + a_{30}) \\ &= a_{25} + a_{30} \end{aligned}$$

Concentrating on *triangles*, he finds that the meaning of the sentence ‘*No triangles are blue*’ computes to $\text{not} \circ \text{blue} \circ \text{inc}_{c,n} \circ \text{triangles}$ in the lexical category, by (2). Hence, the interpretation of the sentence

in the functional vector space model \mathcal{F} maps logical words to logical connectives and the inequalities to identities. Therefore $\mathcal{F}(\text{not}) = \text{not}$ and the meaning of the sentence in the model \mathcal{F} is

$$\text{not} \circ \mathcal{F}(\text{blue}) \circ \mathcal{F}(\text{triangle}) = \text{not}(10 \cdot \perp) = 10 \cdot \top.$$

The resulting vector is colinear to \top , hence the sentence *No triangles are blue* is true in the model. The Fundamental Property implies that $\mathcal{F}(\text{blue})(x) = \perp$ for every basis vector $x \in \mathcal{F}(\text{triangle})$. The predicates $\mathcal{F}(\text{red})$ and $\mathcal{F}(\text{yellow})$ are mixed on $\mathcal{F}(\text{triangle})$

$$\begin{aligned} \mathcal{F}(\text{red} \circ \text{triangle}) &= \mathcal{F}(\text{red})(a_1 + \dots + a_9) + \mathcal{F}(\text{red})(a_{10}) = 9 \cdot \top + 1 \cdot \perp \\ \mathcal{F}(\text{yellow} \circ \text{triangle}) &= 4 \cdot \top + 6 \cdot \perp. \end{aligned}$$

Example 2. *Property? : probability of yes*

The player decides to describe the concepts *triangle*, *square*, *circle* by their colours and use the probability that a chip with colour combination c_j has shape p . Therefore he needs the number k_{pj} of chips of shape p appearing with a given colour combination c_j and the number m_j of chips having colour combination c_j . Using r if the red colour is present and $\neg r$ if the red colour is absent and similarly for the other colours, he first arranges the chips according to their colour combinations

combination	chips	number
$c_1 = r \ \neg y \ b$	$C_1 = \{a_{16}, \dots, a_{20}\}$	$m_1 = 5$
$c_2 = r \ y \ \neg b$	$C_2 = \{a_7, a_8, a_9\} \cup \{a_{11}, \dots, a_{15}\}$	$m_2 = 8$
$c_3 = r \ \neg y \ \neg b$	$C_3 = \{a_1, \dots, a_6\}$	$m_3 = 6$
$c_4 = \neg r \ y \ \neg b$	$C_4 = \{a_{10}, a_{21}\}$	$m_4 = 2$
$c_5 = \neg r \ \neg y \ \neg b$	$C_5 = \{a_{22}, \dots, a_{30}\}$	$m_5 = 9$.

number	shape	combination
$k_{s1} = 5$	$s = \text{square}$	$c_1 = r \ \neg y \ b$
$k_{s2} = 5$	$s = \text{square}$	$c_2 = r \ y \ \neg b$
$k_{c4} = 1$	$c = \text{circle}$	$c_4 = \neg r \ y \ \neg b$
$k_{c5} = 9$	$c = \text{circle}$	$c_5 = \neg r \ \neg y \ \neg b$
$k_{t3} = 6$	$t = \text{triangle}$	$c_3 = r \ \neg y \ \neg b$
$k_{t2} = 3$	$t = \text{triangle}$	$c_2 = r \ y \ \neg b$
$k_{t4} = 1$	$t = \text{triangle}$	$c_4 = \neg r \ y \ \neg b$

Representing each shape p by the extent to which the primitive properties are true, namely the vector

$$p = \sum_j k_{pj}/m_j \cdot c_j$$

he obtains

$$\begin{aligned} \mathcal{F}(\text{square}) &\mapsto \text{square} = 5/5 \cdot c_1 + 5/8 \cdot c_2 \\ \mathcal{F}(\text{triangle}) &\mapsto \text{triangle} = 3/8 \cdot c_2 + 6/6 \cdot c_3 + 1/2 \cdot c_4 \\ \mathcal{F}(\text{circle}) &\mapsto \text{circle} = 1/2 \cdot c_4 + 9/9 \cdot c_5. \end{aligned}$$

It suffices to ask for a red chip that is not yellow and not blue to obtain a triangle.

5 Conceptual vector semantics

Distributional vector space models represent words by vectors in a finite dimensional space V . Its basis vectors are assimilated with previously chosen words, e.g. the key words of a thesaurus or the most frequent words in a set of documents. If the distribution is based on context, the coordinates of a word vector stand for frequencies of co-occurrences with the respective basis vectors in contexts.

This study takes a more general approach. We assume that the coefficients of the word vectors belong to the real interval $[0, 1]$. Call *concept vector* any vector that has its coordinates in $[0, 1]$. The method by which the coordinates have been obtained is irrelevant in this section. We want to define a logic on the space V so that we can reason with vectors in a vector space model according to our intuition. For instance, the vectors assigned to the statements *All apples are juicy* and *No apples are juicy* must be contradictory.

The following definitions refer to an orthonormal basis $A = \{a_1, \dots, a_n\}$ of \mathbb{R}^n . A vector is *Boolean* if its coordinates with respect to A are equal to 0 or 1. The set of Boolean vectors is denoted

$$\mathcal{B}_A = \left\{ \sum_{i=1}^n \alpha_i a_i : \alpha_i \in \{0, 1\} \right\}.$$

Denote $\lceil f \rceil_{AA}$ the matrix in the basis A defined by an endomorphism f . The endomorphisms of \mathbb{R}^n that are diagonalisable with respect to A and the projectors among them form the sets

$$\begin{aligned} \mathcal{D}_A &= \{f : \mathbb{R}^n \longrightarrow \mathbb{R}^n : \lceil f \rceil_{AA} \text{ is a diagonal matrix}\} \\ \mathcal{P}_A &= \{f \in \mathcal{D}_A : f \circ f = f\}. \end{aligned}$$

There is an obvious bijection between vectors of \mathbb{R}^n and \mathcal{D}_A via the correspondence

$$X = \sum_{i=1}^n \alpha_i a_i \leftrightarrow D_X = \begin{pmatrix} \alpha_1 & & 0 \\ & \ddots & \\ 0 & & \alpha_n \end{pmatrix}$$

This correspondence is the vector space analogue of the bijection between subsets and predicates in set theory. Moreover, it maps \mathcal{B}_A onto \mathcal{P}_A .

The algebraic connectives below generalise the connectives of predicate logic and coincide with those of quantum logic for projectors arising in vector space models. They are inspired by a probabilistic interpretation of truth.

ALGEBRAIC CONNECTIVES

The *algebraic connectives* are defined for scalars and for arbitrary square matrices thus

$$\begin{array}{llll} \text{negation} & \neg \alpha & = 1 - \alpha & \neg D & = 1 - D \\ \text{conjunction} & \alpha \wedge \beta & = \alpha \beta & D \wedge E & = DE \\ \text{disjunction} & \alpha \vee \beta & = \alpha + \beta - \alpha \beta & D \vee E & = D + E - DE \\ \text{implication} & \alpha \rightarrow \beta & = 1 - \alpha + \alpha \beta & D \rightarrow E & = 1 - D + DE \end{array} .$$

The algebraic connectives are lifted from scalars to vectors by the conditions

$$\begin{aligned} \neg X &= \neg(\sum_{i=1}^n \alpha_i a_i) &= \sum_{i=1}^n (\neg \alpha_i) a_i \\ X \vee Y &= (\sum_{i=1}^n \alpha_i a_i) \vee (\sum_{i=1}^n \beta_i a_i) &= \sum_{i=1}^n (\alpha_i \vee \beta_i) a_i, \end{aligned}$$

where ∇ stands for any of the binary algebraic connectives. It follows that

$$\neg D_X = D_{\neg X} \quad \text{and} \quad D_X \nabla D_Y = D_{X \nabla Y} \quad (6)$$

The two equalities above say that the one-to-one correspondence that identifies a vector X with the diagonal matrix D_X is an isomorphism. Note that the conjunction of two vectors is the same as their pointwise product and that they relate to composition by the equalities

$$D_X \circ D_Y = D_X \wedge D_Y = D_{X \wedge Y} = D_{X \odot Y}, \quad D_X \circ \langle Y | = \langle X \wedge Y |, \quad (7)$$

where $\langle Z |$ denotes the matrix of the linear map from I to V_A that assigns the vector $Z \in V_A$ to the basis vector of I .

Concept vectors are closed under the algebraic connectives, because the interval $[0, 1]$ is closed under the algebraic connectives on scalars. The proof is straightforward. For instance, to show the assertion for the algebraic disjunction, assume $\alpha, \beta \in [0, 1]$. The inequality $0 \leq \alpha + \beta - \alpha\beta$ follows from $\beta - \alpha\beta = \beta(1 - \alpha) \geq 0$. The inequality $\alpha + \beta - \alpha\beta \leq 1$ follows from $1 - \alpha - \beta + \alpha\beta = (1 - \alpha)(1 - \beta) \geq 0$.

The algebraic connectives do not define a lattice structure, the algebraic conjunction for instance is not idempotent unless the involved scalars are equal to 0 or 1. The algebraic connectives have, however, several properties with a logical flavour, among them the laws of a weak conditional logic in the sense of (13).

Any real numbers $\alpha, \beta \in [0, 1]$ and diagonal matrices $D, E \in \mathcal{D}_A$ with entries in $[0, 1]$ satisfy

$$\alpha \rightarrow \beta = 1 \iff \alpha = 0 \text{ or } \beta = 1 \quad (8)$$

$$\alpha \rightarrow \beta = 1 \iff \alpha\beta = \alpha$$

$$\text{If } \alpha \rightarrow \beta = 1 \text{ then } \alpha \leq \beta$$

$$D \rightarrow E = 1 \iff E \circ D = D$$

$$\text{If } D \rightarrow E = 1 \text{ then } D \leq E.$$

ALGEBRAIC CONSEQUENCE RELATION

The endomorphism defined by E is an *algebraic consequence* of that defined by D if and only if

$$D \rightarrow E = 1.$$

PROBABILISTIC CONSEQUENCE RELATION

The endomorphism defined by E is a *probabilistic consequence* of that defined by D if

$$D \leq E.$$

Projectors stand for properties in quantum logic and geometrical operations define the connectives. These geometrical operations are introduced in (13) via the range of the involved projectors based on the fact that for every subspace there is a unique projector which maps the whole space onto the subspace in question.

GEOMETRICAL CONNECTIVES

Let $p, q : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be projectors. Then

the *geometrical negation* $\neg p$ is the unique projector that has range

$$(\neg p)(\mathbb{R}^n) = p(\mathbb{R}^n)^\perp$$

the *geometrical conjunction* $p \wedge q$ is the unique projector that has range

$$(p \wedge q)(\mathbb{R}^n) = p(\mathbb{R}^n) \cap q(\mathbb{R}^n)$$

the *geometrical disjunction* $p \vee q$ is the unique projector that has range

$$(p \vee q)(\mathbb{R}^n) = p(\mathbb{R}^n) + q(\mathbb{R}^n)$$

the *geometrical implication* $p \Rightarrow q$ is the unique projector that has range

$$(p \Rightarrow q)(\mathbb{R}^n) = \{x \in \mathbb{R}^n : (q \circ p)(x) = p(x)\}.$$

QUANTUM CONSEQUENCE RELATION

Projector q is said to be a *geometrical consequence* of projector p if and only if

$$p \Rightarrow q = 1.$$

The definition makes the detour via the subspaces, because there is no obvious algebraic operation defining the projector. For example, $p \circ q$ maps \mathbb{R}^n onto the intersection of the image of p and the image of q , but $p \circ q$ is not a projector unless p and q commute. If p and q do not commute, there is no basis in which they are both diagonalisable. Are we not losing representatives of properties in probability when replacing the projectors by \mathcal{D}_A ? The answer is that to the contrary, we are gaining representatives at least as long as we accept the geometric consequence relation.

Proposition 1. *If projector q is a geometrical consequence of projector p then there is an orthogonal basis of \mathbb{R}^n consisting of eigenvectors of both p and q .*

The geometrical consequence relation and the algebraic consequence relation coincide on projectors. If one of p or q is a geometrical consequence of the other then the geometrical connectives coincide with the algebraic connectives for p and q .

Proof. (Outline) Clearly, the second statement follows from the first. To see the first statement, assume that q is a geometrical consequence of p and let A be an orthonormal basis of \mathbb{R}^n formed by eigenvectors of p . The eigenvectors in A left invariant by p are also left invariant by q . Let A_1 denote this set and V be the subspace generated by A_1 . Then q maps the orthogonal complement of V onto itself whereas p maps it to 0. Hence any set C of orthonormal eigenvectors of q belonging to V are also eigenvectors of p . Thus $A_1 \cup C$ is a basis of orthonormal eigenvectors for both p and q . \square

The proposition above also implies that the algebraic connectives can be captured by geometrical properties, at least on Boolean vectors. In particular, two distinct basis vectors contradict each other. This raises the question of how to choose the basis vectors so that they represent contradictory properties.

6 Distributional interpretations

Everyday language switches commonly from asserting facts about some real or possible world to updating the concepts intervening in the statements about the facts. This switch is related to the canonical distribution based on the counting property of predicates. The coefficient of a property, say *apple*, at a basis vector, say *juicy*, is the probability of the event *apple* given the event *juicy*. It gives us the extent to which the property *juicy* is characteristic for the concept *apple*.

The section concludes with sufficient conditions for concept logic to be reflected and predicate logic to be preserved.

Predicates q_1, \dots, q_k on $A = \{a_1, \dots, a_n\}$ are said to *partition* A if for any $a \in A$ there is a j for which $q_j(a) = \top$ and $i \neq l$ implies $q_i(a) = \perp$ or $q_l(a) = \perp$ for $i, l = 1, \dots, k$. Clearly, families of partitioning predicates are in one-to-one correspondence with set-theoretical partitions $\{C_1, \dots, C_k\}$ of A given by

$$C_j = \{a \in A : q_j(a) = \top\}, \quad j = 1, \dots, k.$$

Let $m_j = |C_j|$ so that $\sum_j m_j = n = |A|$. Assuming that every individual in A has probability $1/n$, the real number

$$\mu_j = m_j/n$$

can be understood as the probability that an arbitrary individual $a \in A$ has property q_j , for $j = 1, \dots, k$.

Choose some orthonormal basis $C = \{c_1, \dots, c_k\}$ of $V_C = \mathbb{R}^k$. Think of the basis vectors c_1, \dots, c_k as ‘basic events on A ’ or as ‘basic properties’ of the elements of A . The density operator defined by the diagonal matrix

$$D_\mu = \begin{pmatrix} \mu_1 & & 0 \\ & \ddots & \\ 0 & & \mu_k \end{pmatrix}$$

summarises the first order model consisting of A and the predicates q_j , for $j = 1, \dots, k$. The composite of the density operator D_μ with the j ’th projector of \mathbb{R}^k maps an arbitrary vector $\sum_{i=1}^k \beta_i c_i$ to $\mu_j \beta_j$. We shall define for any property p on A a vector $\mathcal{J}(p) = \sum_{i=1}^k \alpha_i c_i \in V_C$ such that $\sum_{j=1}^k \alpha_j \mu_j = n_{pA}/n$. Recall that n_{pA} is the number of elements of A satisfying p . If we call p a ‘state’ of the ‘system’ A , q_1, \dots, q_k then n_{pA}/n is the probability that the system is in state p .

For any predicate p on A , the integer n_{pC_j} is the coefficient of $p(\vec{C}_j)$ at \top . Define

$$\mathcal{J}_C(p) = \sum_{j=1}^k \alpha_{pj} c_j, \quad \text{where } \alpha_{pj} = \begin{cases} n_{pC_j}/m_j & \text{if } m_j \neq 0 \\ 0 & \text{else} \end{cases}, \quad \text{for } j = 1, \dots, k.$$

The number α_{pj} is the conditional probability that an element has property p given property q_j . It follows from the linearity of p that $n_{pA} = \sum_j n_{pC_j}$. Therefore, the probability that an arbitrary element of A has property p is equal to

$$n_{pA}/n = \sum_j n_{pC_j}/n = \sum_j (n_{pC_j}/m_j)(m_j/n) = \text{trace}(D_\mu \circ D_{\mathcal{J}(p)}) = \sum_j \alpha_{pj} \mu_j.$$

The interpretation \mathcal{J}_C is a one-to-one map from predicates on A to Boolean vectors of $V_C \simeq V_A$ if C consists of the singleton sets $\{a_1\}, \dots, \{a_n\}$. If this is the case then $\alpha_{pj} = 1$ if $p(a_j) = \top$ and $\alpha_{pj} = 0$ otherwise. In the general case, \mathcal{J}_C is neither one-to-one nor onto the set of vectors with coefficients in $[0, 1]$. The following Lemma describes the general situation.

Lemma 1. *Let $p, q : V_A \rightarrow S$ be any predicates on A , assume that the sets C_j are not empty for $j = 1, \dots, k$ and that for every j at least one of p or q is constant on C_j . Then the following holds*

$$\begin{aligned} \mathcal{J}_C(\text{and} \circ \langle p, q \rangle) &= \mathcal{J}_C(p) \wedge \mathcal{J}_C(q) \\ \mathcal{J}_C(\text{or} \circ \langle p, q \rangle) &= \mathcal{J}_C(p) \vee \mathcal{J}_C(q) \\ \mathcal{J}_C(\text{ifthen} \circ \langle p, q \rangle) &= \mathcal{J}_C(p) \rightarrow \mathcal{J}_C(q). \end{aligned}$$

Theorem 1. *Suppose that the non-empty sets C_1, \dots, C_k partition A and that p and q are predicates on A . Then \mathcal{I}_C preserves negation*

$$\mathcal{I}_C(\text{not} \circ p) = \neg \mathcal{I}_C(p)$$

and reflects the consequence relation

$$\mathcal{I}_C(p) \rightarrow \mathcal{I}_C(q) = 1 \text{ implies } \text{ifthen} \circ \langle p, q \rangle = \text{true} \quad (9)$$

If one of $\mathcal{I}_C(q)$ and $\mathcal{I}_C(p)$ is an algebraic/geometrical consequence of the other then \mathcal{I}_C preserves the logical connectives, i.e. the algebraic/geometrical connectives preserve the probabilistic interpretation of concept vectors.

$$\begin{aligned} \mathcal{I}_C(p) \wedge \mathcal{I}_C(q) &= \mathcal{I}_C(\text{and} \circ \langle p, q \rangle) \\ \mathcal{I}_C(p) \vee \mathcal{I}_C(q) &= \mathcal{I}_C(\text{or} \circ \langle p, q \rangle) \\ \mathcal{I}_C(p) \rightarrow \mathcal{I}_C(q) &= \mathcal{I}_C(\text{ifthen} \circ \langle p, q \rangle) \\ \mathcal{I}_C(q) \rightarrow \mathcal{I}_C(p) &= \mathcal{I}_C(\text{ifthen} \circ \langle q, p \rangle). \end{aligned} \quad (10)$$

Proof. Assume that $\mathcal{I}_C(p) \rightarrow \mathcal{I}_C(q) = 1$. Then $1 - \alpha_{pj} + \alpha_{pi}\alpha_{qj} = 1$ and therefore $\alpha_{pj} = 0$ or $\alpha_{qj} = 1$, for $j = 1, \dots, k$. Otherwise said, p maps every element of C_j to \perp or q maps every element of C_j to \top . As every element of A belongs to some C_j , we have

$$\text{ifthen} \circ \langle p, q \rangle(x) = \top, \text{ for all } x \in A.$$

The equality $\text{ifthen} \circ \langle p, q \rangle = \text{true}$ follows. This completes the proof of (9).

The equalities (10) hold, because the assumptions of the preceding lemma are satisfied. \square

Negation is preserved exactly when D_μ is positive definite, i.e $\mu_j > 0$, for $j = 1, \dots, k$. This condition alone is not sufficient for \mathcal{I}_C to preserve the binary connectives.

In the particular case where the partition of A is $C_1 = \{a_1\}, \dots, C_n = \{a_n\}$, the hypotheses of Lemma 1 and Theorem 1 are satisfied. Therefore, \mathcal{I}_A is an isomorphism of the Boolean algebra of predicates on A onto the Boolean algebra of Boolean vectors \mathcal{B}_A of V_A . Hence in the case where $\mu_i = 1/n$, for $i = 1, \dots, n$, predicate logic, quantum logic and vector space logic are the same, because the lattice of predicates on A , the lattice of projectors in \mathcal{P}_A and the lattice of Boolean vectors \mathcal{B}_A are isomorphic. Composition of projectors, conjunction of predicates and pointwise product of vectors are three variants of the same operation on property words.

In the general case, extend \mathcal{I}_C from predicates on A to projectors and Boolean vectors of the space V_A . Then Theorem 1 remains valid. Moreover, the composite $\mathcal{I}_C \circ \mathcal{F}$ is a well defined map from property words to vectors in V_C . The induced compact closed structure preserving functor $\mathcal{M}_C : \mathcal{L}(\mathcal{B}) \rightarrow V_C$ given by

$$\mathcal{M}_C(\text{word}_T) = \mathcal{I}_C(\mathcal{F}(\text{word}_T))$$

maps the lexical meaning of a string to the pointwise product of the word vectors, by Equation (3). The truth-probabilistic relation between the vector model and the logical functional model extends under the sufficient conditions of Theorem 1 to string of words

$$\mathcal{M}_C(\text{word}_{1T_1}) \odot \dots \odot \mathcal{M}_C(\text{word}_{nT_n}) = \mathcal{I}_C(\mathcal{F}(r \circ (\overline{\text{word}}_{1T_1} \otimes \dots \otimes \overline{\text{word}}_{nT_n}))).$$

Theorem 1 also provides a method for checking how appropriate a possible extension of an arbitrary vector model to other noise words would be. For example, $\mathcal{F}(\text{are})$ is the identity map in all functional

models. Hence $\mathcal{F}(\text{rocks are grey}) = \mathcal{F}(\text{grey}) \circ \mathcal{F}(\text{rocks})$. The unit for \odot , the vector $\vec{1}$, is the identity when we think of \odot as composition. Hence we let $\mathcal{M}(\text{are}) = \vec{1}$ so that in the vector model

$$\mathcal{M}(\text{rocks}) \odot \mathcal{M}(\text{are}) \odot \mathcal{M}(\text{grey}) = \mathcal{M}(\text{rocks}) \odot \mathcal{M}(\text{grey}) = \mathcal{M}(\text{rocks}) \wedge \mathcal{M}(\text{grey}).$$

Similarly, if $\mathcal{M}(\text{and}) = \vec{1}$ then \mathcal{M} maps the lexical meaning $\text{and} \circ (\text{red} \otimes \text{blue}) : \mathbf{p} \otimes \mathbf{p} \rightarrow \mathbf{p}$ of the string *red and blue* to

$$\mathcal{M}(\text{and}) \odot \mathcal{M}(\text{red}) \odot \mathcal{M}(\text{blue}) = \mathcal{M}(\text{red}) \odot \mathcal{M}(\text{blue}) = \mathcal{M}(\text{red}) \wedge \mathcal{M}(\text{blue}).$$

Our toy example concerns unary predicates only, for the sake of simplicity, but the definitions apply to arbitrary predicates. Consider the case of binary predicates. A logical functional model \mathcal{F} interprets a transitive verb as a binary predicate $p : V_{E \otimes E} \simeq V_E \otimes V_E \rightarrow S$ and nouns as vectors in V_E . A somewhat realistic pregroup lexicon lists a transitive verb with type $\mathbf{n}_{\text{sub}}^r \mathbf{sn}_{\text{ob}}^\ell$ and a noun with the types \mathbf{n}_{sub} and \mathbf{n}_{ob} . Note that the tensor product of vectors $v, w \in V_A$ is related to the pointwise product in $V_{E \otimes E}$ by the equality

$$v \otimes w = (v \otimes \vec{1}) \odot (\vec{1} \otimes w)$$

The passage from \mathcal{F} to the vector space model \mathcal{M} described above is facilitated by defining

$$M(\text{noun} : \mathbf{n}_{\text{sub}}) = v \otimes \vec{1} \quad M(\text{noun} : \mathbf{n}_{\text{ob}}) = \vec{1} \otimes w,$$

where $v = \mathcal{I}(\mathcal{F}(\text{noun}_{\mathbf{n}_{\text{sub}}}))$ and $w = \mathcal{I}(\mathcal{F}(\text{noun}_{\mathbf{n}_{\text{ob}}}))$.

The fact that the binary operation \odot is commutative does not imply that meanings are necessarily commutative, because the model interprets the sentence *cats chase dogs* by

$$\mathcal{M}(\text{cats}_{\mathbf{n}_{\text{sub}}}) \odot \mathcal{M}(\text{chase}_{\mathbf{n}_{\text{sub}} \mathbf{sn}_{\text{ob}}^\ell}) \odot \mathcal{M}(\text{dogs}_{\mathbf{n}_{\text{ob}}}),$$

and *dogs chase cats* by

$$\mathcal{M}(\text{dogs}_{\mathbf{n}_{\text{ob}}}) \odot \mathcal{M}(\text{chase}_{\mathbf{n}_{\text{sub}}^r \mathbf{sn}_{\text{ob}}^\ell}) \odot \mathcal{M}(\text{cats}_{\mathbf{n}_{\text{sub}}}).$$

7 Logic and the basis of the vector space

Theorem 1 in the preceding section shows that the quality of reasoning in a vector space model depends essentially on the choice of the basic concepts. The most frequent property words in a document do not in general constitute a partition. In the world of fruit, there may be things that are *juicy* and *sweet* simultaneously. It is the requirement borrowed from quantum logic that orthogonal vectors must be contradictory that forces basis vectors to be contradictory.

There is however a general method for transforming an arbitrary choice of properties into a set of partitioning properties. This method was used in our Example 2.

Let $P = \{w_1, \dots, w_d\}$ be a set of property words in the lexicon. Think of them as primitive properties. Invent a two-dimensional space $S_i = V_{\{w_i, \neg w_i\}}$ with basis vectors w_i and $\neg w_i$, for $i = 1, \dots, d$ and define the *concept space generated by P* as

$$C(P) = S_1 \otimes \dots \otimes S_d.$$

The basis vectors of $C(P)$ are of the form

$$c_j = c_j(1) \otimes \dots \otimes c_j(d)$$

where $c_j(i) \in \{w_i, \neg w_i\}$, for $j = 1, \dots, 2^d$, $i = 1, \dots, d$.

Without loss of generality, we may assume that the functional model interprets the words w_1, \dots, w_d as predicates p_1, \dots, p_d on a space V_A . The basis vectors correspond to the following partition C_1, \dots, C_{2^d} of subsets of A

$$a \in C_j \iff p_i(a) = \begin{cases} \top & \text{if } c_j(i) = w_i \\ \perp & \text{if } \neg w_i \end{cases}, \text{ for } i = 1, \dots, d.$$

If we work within the subspace C of $C(P)$ generated by the basis vectors c_j for which $C_j \neq \emptyset$, the interpretation \mathcal{I}_C reflects concept logic. It preserves predicate logic under the conditions of Theorem 1.

Example 3.

A text describing the world of coloured chips may contain pertinent knowledge about the relation of shapes and colours. The concept triangle to be characterised by the three primitive properties *red*, *yellow* and *blue* has the form

$$\text{triangle} = \alpha_1 c_1 + \dots + \alpha_8 c_8, 0 \leq \alpha_i \leq 1.$$

The truth of the statement *No triangle is blue* implies that $\alpha_i = 0$ for all four colour combinations c_i involving b (blue) without the negation symbol. Hence

$$\text{triangle} = \alpha_2 c_2 + \alpha_3 c_3 + \alpha_4 c_4 + \alpha_5 c_5.$$

Note that the vector *triangle* is orthogonal to every basis vector that lists the colour blue as present, namely c_1, c_6, c_7 and c_8 . Therefore *triangle* is orthogonal to the subspace ‘blue’ generated by c_1, c_6, c_7 and c_8 .

The player wants to find out if there is also a winning strategy for a new square. He computes the concept corresponding to the predicate $\mathcal{F}(\text{new})$ and then the concept *new square*

$$\begin{aligned} \mathcal{I}(\mathcal{F}(\text{new})) &= 1/5 \cdot c_1 + 8/8 \cdot c_2 + 1/6 \cdot c_3 + 1/2 \cdot c_4 + 2/9 \cdot c_5 \\ \mathcal{I}(\mathcal{F}(\text{new} \circ \text{square})) &= 1/5 \cdot c_1 + 5/8 \cdot c_2 = \mathcal{I}(\mathcal{F}(\text{new})) \odot \mathcal{I}(\mathcal{F}(\text{square})). \end{aligned}$$

He guesses that the same works for the other shapes. But

$$\mathcal{I}(\mathcal{F}(\text{new} \circ \text{triangle})) = 3/8 \cdot c_2 + 1/6 \cdot c_3 + 1/2 \cdot c_4 \neq \mathcal{I}(\mathcal{F}(\text{new})) \odot \mathcal{I}(\mathcal{F}(\text{triangle})).$$

In our particular characterisation of the world of chips the concept ‘new’ does not always interact logically with the other concepts.

8 Conclusion

The preceding is only an outline how to extend vector space models compositionally to statements that go beyond the property words. The vector space models depend on the chosen probability on the vector space standing for the ‘universe of discourse’. Grammar is another parameter of our vector space models. Different types may result in different meanings. The parameter ‘background knowledge’ is also present via the choice of the primitive properties leading to the basis vectors of the concept space.

The truth-probabilistic approach to compositional vector space models provides a tool to compare probabilistic reasoning in vector space models and reasoning with traditional logical tools, depending on the parameters.

References

- [1] Nicholas Asher (2011): *Lexical Meaning in Context*. Cambridge University Press.
- [2] Stephen Clark, Bob Coecke & Mehrnoosh Sadrzadeh (2008): *A Compositional Distributional Model of Meaning*. In W. Lawless P. Bruza & J. van Rijsbergen, editors: *Proceedings of Conference on Quantum Interactions*, University of Oxford, College Publications.
- [3] Edward Grefenstette & Mehrnoosh Sadrzadeh (2011): *A Compositional Distributional Semantics Two Concrete Constructions and some Experimental Evaluations*. In: *Lecture Notes in Computer Science*, Springer. Pending publication.
- [4] Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, Stephen Pulman & Bob Coecke (2013): *Reasoning about Meaning in Natural Language with Compact Closed Categories and Frobenius Algebras*. Cambridge University Press.
- [5] G.M. Kelly & M.L. Laplaza (1980): *Coherence for compact closed categories*. *Journal of Pure and Applied Algebra* 19, pp. 193–213.
- [6] Marcus Kracht (2007): *Compositionality: The Very Idea*. *Research in Language and Computation* 5, pp. 287–308.
- [7] Joachim Lambek (1999): *Type Grammar revisited*. In Alain Lecomte, editor: *Logical Aspects of Computational Linguistics*, LNAI 1582, Springer, Heidelberg, pp. 1–27.
- [8] Joachim Lambek (2008): *From word to sentence*. Polimetrica, Milano, Italia.
- [9] Saunders Mac Lane (1971): *Categories for the Working Mathematician*. Springer.
- [10] Jeff Mitchell & Mirella Lapata (2008): *Vector-based Models of Semantic Composition*. In: *Proceedings of the 46th Annual Meeting of Computational Linguistics*, pp. 236–244.
- [11] Anne Preller & Joachim Lambek (2007): *Free compact 2-categories*. *Mathematical Structures for Computer Sciences* 17(1), pp. 1–32, doi:<http://dx.doi.org/10.1017/S0960129506005901>.
- [12] Anne Preller & Mehrnoosh Sadrzadeh (2011): *Semantic Vector Models and Functional Models for Pregroup Grammars*. *Journal of Logic, Language and Information* 20(4), pp. 419–423, doi:<http://dx.doi.org/10.1007/s10849-011-9132-2>.
- [13] C.J. van Rijsbergen (2004): *The Geometry of Information Retrieval*. Cambridge University Press.
- [14] Dominic Widdows (2004): *Geometry and Meaning*. *CSLI lecture notes 172*, CSLI Publications.
- [15] Dominic Widdows (2008): *Orthogonal negation in vector-spaces for modelling word-meanings and document retrieval*. In: *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*.