# Technical Proposal

## Algorithmic and Logical Aspects when Composing Meanings

Words are the building blocks of sentences, yet meaning of a sentence goes well beyond meanings of words therein. Indeed, while we do have dictionaries for words, we don't seem to need them to infer the meaning of a sentence from meanings of its constituents. Discovering the process of meaning assignment in natural languages is one of the most foundational issues in linguistics and computer science, whose findings will assist in crafting applications to automate many language-related tasks, such as document search, automated text generation, an many others.

To date, the compositional type-logical [16, 14] and the distributional probabilistic models [17, 9] have provided two complementary partial solutions to the problem of meaning assigning in natural languages. The logical approach is based on classical ideas from mathematical logic, mainly Frege's principle that meaning of a sentence can be derived from the relations of the words in it. The distributional model is more recent, it can be related to Wittgenstein's philosophy of 'meaning as use', whereby meanings of words can be determined from their context. The logical models have been the champions on the theory side, whereas in practice their probabilistic rivals have provided the best predictions.

This two-sortedness of defining properties of meaning: 'logical form' versus 'contextual use', has left the question of 'what is the foundational structure of meaning?' even more open a question than before.

A breakthrough towards this goal was achieved by the PIs of this proposal, by proposing a compositional distributional model of meaning that combines the compositional type-logical and the distributional probabilistic models (hyperlink):

`Mathematical Foundations for a Compositional Distributional Model of Meaning (Coecke, Sadrzadeh and Clark)`

This work has been conceived as groundbreaking and was also covered by the popular media, e.g. New Scientist and Scientific American (hyperlinks):

`http://www.newscientist.com/article/mg20827903.200-quantum-links-lters-understand-language.html`

`http://blogs.scientificamerican.com/guest-blog/2013/05/16/quantum-mechanical-words-and-mathematical-organisms/`

Moreover, both Sadrzadeh and Clark have been awarded prestigious five-year fellowship on the basis of this work, and a 1.5 M GB research network is now funded by EPSRC which aims for other UK research groups to adopt this model.

## Sketch of the technical backbone

Algebraic gadgets that govern grammatical types have been around for a long time. They have a composition operation that allows to build larger strings of words from smaller strings of words, as well as a relation $\leq$ where $a \cdot \ldots \cdot z \leq t$ means that the string of types $a \ldots z$ has as its overall type $t$. For example, $n \cdot tv \cdot n \leq s$ expresses the fact that a noun, a transitive verb and a noun make up a sentence $s$. Additional operations subject to certain laws allow one to derive correct statements such as $n \cdot tv \cdot n \leq s$, where some types, may take a compound form in terms of other types,. In the case of pregroups we have $tv = {}^{-1}n \cdot s \cdot n^{-1}$ and the derivation can be depicted diagrammatically, and $n \cdot tv \cdot n \leq s$ becomes:


In [1] it was shown that these kinds of diagrams govern vector space calculus, and the passage from compositional type-grammar to compositional distributional meaning build down to interpreting the types as vector spaces in which word-meanings live, and the wire diagrams as linear maps. This leads to an algorithm that allows to compute sentence meaning from word meaning:

1. Perform type reduction:

$$(\textit{word type } 1) \ldots (\textit{word type } n) \;\rightsquigarrow\; \textit{sentence type}$$

2. Interpret diagrammatic type reduction as linear map:

$f ::$  $\mapsto \left(\sum_i \langle ii|\right) \otimes \mathrm{id} \otimes \left(\sum_i \langle ii|\right)$

3. Apply this map to tensor of word meaning vectors:

$$f\left(\overrightarrow{v}_1 \otimes \ldots \otimes \overrightarrow{v}_n\right)$$

A particularly appealing feature of the model is that computations can be done in a purely diagrammatic language where computation rules are simple topology-preserving transformations.

A detailed presentation of the model can be found in [7] and experimental evidence is provided in [12].

## Proposed work

The aimed contribution of this project is to further develop this unified model of meaning, whereby meanings of sentences are built and reasoned about, in a compositional and dynamic way, based on their grammar and the meanings of their constituent words, which themselves are obtained from a practical, natural and robust model.

The results of the project will improve the performance of what has become an inseparable toolkit of our daily lives: the internet with its huge pool of services and naturally occurred data. We aim to achieve these goals alongside the strength of the complementary existing approaches, using scientific methods from different disciplines, including computer science, logic, and physics.

The project has 2 major interconnected strands:

1. To further develop the process of meaning assignment that acts with the compositional forms of the logical model on the contextual word-meaning entities of the distributional model, based on novel information-flow techniques [1], as well as on other linguistic approaches, and other models of word meaning, such as ontological domains [8] and conceptual spaces [10].

2. To evaluate our theories against naturally occurring data and apply the results to practical issues based on meaning inference and similarity, e.g. in search; we develop and implement algorithms to automatize related such tasks, which may ultimately leading to practical tools.

The work will be divided in a number of workpackages.

### W1. Logic and meaning

The overall purpose of this workpackage is to integrate the compositional distributional model of meaning with additional logical structure. Vector spaces (or distributions) are notoriously bad in encoding logical structure. However, with linear logic [11] it became clear that logical structure can be re-instated on top of linear structure by means of additional operations.

#### W1a. Meanings of functional words

Within the context of categorical quantum mechanics , which provide initial inspiration of the compositional distributional model of meaning, certain Frobenius algebras [6] allowed one to represent classical operations. Moreover, they allowed to represent relational structure within linear structure, so since relational structure comes with boolean logic operations, the indicate a manner to represent logical operations within the distributional model of meaning. Initial investigations showed the promise of this method in that the basic relative pronouns were successfully represented, with both in a satisfactory conceptual manner, and supporting with experimental evidence [2]. We aim to extend this to other functional words, and also the logical words ‘and’ and ‘or’, and implications.

### W1b. Word meaning with intrinsic logical structure

As already mentioned above, ‘plain’ vector spaces structure does not accommodate logical structure well. However, density matrices have some logical structure build inside, namely conjunction, disjunction and conjugation, and is still subject to the high-level categorical formalism [18, 3]. For example, negation of a density matrix $\rho$ is simply obtained as $1 - \rho$ while for commuting operators $\rho, \rho'$ the conjunction is the sum $\rho + \rho'$, and hence conjunction arise via the De Morgan rule. Each vector can be represented as a density matrix via:

$$|v\rangle \mapsto |v\rangle\langle v|$$

Moreover, the proper density matrices (i.e. not of the form $|v\rangle\langle v|$) give useful extra degrees of freedom: they allow for a notion of ‘informative content’ of sentences in that say, the maximally mixed state provides no information whatsoever, while those of the form $|v\rangle\langle v|$ are maximally informative.

These density matrices moreover admit a domain-theoretic structure [5], and hence subjects these to corresponding methods [15].

### W1c. Meanings for paragraphs

Once logical operations are at hand, one can chain sentences by the logical conjunctives such as ‘and’, ‘or’, and build simple text fragments that can also be analysed by the compositional distributional method. As very simple compositional conjunctive operators, consider point wise addition and multiplication. These have been applied to composing sentences with each other in tasks such as tagging dialogue utterances. In such a setting, the meaning of a text containing two sentences $s_1$ and $s_2$, may be represented as follows

$$\overrightarrow{s_1\ s_2} \ = \ \overrightarrow{s}_1 + \overrightarrow{s}_2 \qquad \overrightarrow{s_1\ s_2} \ = \ \overrightarrow{s}_1 \odot \overrightarrow{s}_2$$

But these operations do not reflect the order of sentences and as a result we will obtain the following unwanted consequence:

$$\overrightarrow{s_1\ s_2} = \overrightarrow{s_2\ s_1}$$

This order does makes a difference in the meaning of a paragraph and the text as a whole. In this work package we seek new operations between sentence vectors representing the flow of meaning within paragraphs of a text. An example of such an operations is the tensor product or a convoluted version of it that does not increase the dimensionality. In this case we obtain an order-preserving representation, that is

$$\overrightarrow{s_1\, s_2} \;=\; \overrightarrow{s}_1 \otimes \overrightarrow{s}_2 \;\neq\; \overrightarrow{s}_2 \otimes \overrightarrow{s}_1$$

We will also opt for and work with newer order-preserving versions of conjunction and disjunction, similar to linear logic additives and multiplicatives. The resulting vectors of these methods for simple text fragments can be subjected to the same experimental methods that were used to validate the algorithm to compute sentence meaning from word meaning.

### W2. A calculus of meaning similarity

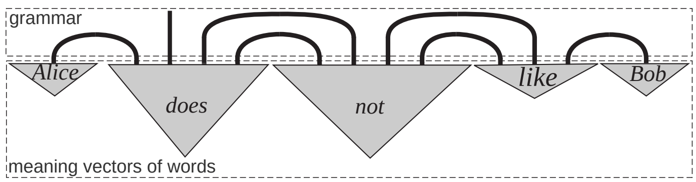
One of the most successful and wide spread applications of the distributional models is formalising word similarity. Once word vectors are built, the distances between them are computed using a variety of measures, each reflecting a different degree of similarity between meanings of word. The aim of this work package is to extend these methods from words to sentences. We would like to be able to reason about similarity of meanings between sentences in a compositional way and develop a calculus of meaning. Such a calculus of meaning will aim for inferring that, for example, if two sentences $s$ and $s'$ each consist of three words $w_1w_2w_3$ and $w'_1w'_2w'_3$, then $s$ and $s'$ have similar meanings whenever we have that $w_i$ has a similar meaning to $w'_i$.

$$s \sim s' \quad \Leftrightarrow \quad w_i \sim w'_i \quad \text{for all } i$$

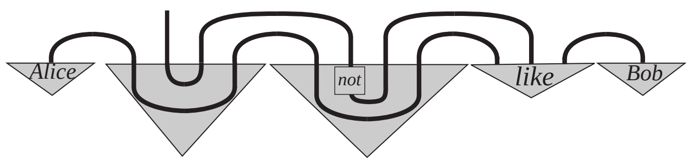
The challenge in such an approach is that according to the experimental results, human beings do not assign equal weights to all words of a sentence. For instance, the two sentences "father bought horse" and "uncle purchased cookie" are considered to be far more similar to each other than the two sentences "father bought horse" and "uncle rode stallion", whereas both pairs of sentences have the same number of pairs of similar words (2 pairs each). The difference in judgement stems from the fact that in the first pair the verbs (bought and purchased) are similar and in the second they are not (rode and bought). In this work package, we aim to enrich the notion of sentence similarity with weights assigned according to the grammatical roles of the words in the sentence.

**W2a. Equations for functional words**

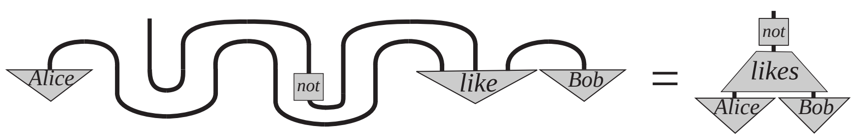
Several functional words can be reduced to other functional words and this can be used to design meanings for more complicated functional words from the simpler ones. In previous work this was implicit and making it explicit led to the development of meanings for a new range of functional words. For instance, in previous work, the meaning of 'does' and 'not' are reverse-engineered from the intended overall meaning of the sentence [7]. Applying the sketched algorithm of page 2, the meaning of the sentence 'Alice does not like Bob' becomes as follows:



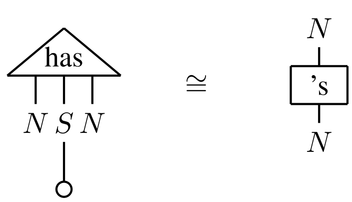

When substituting 'does' and 'not' as follows:



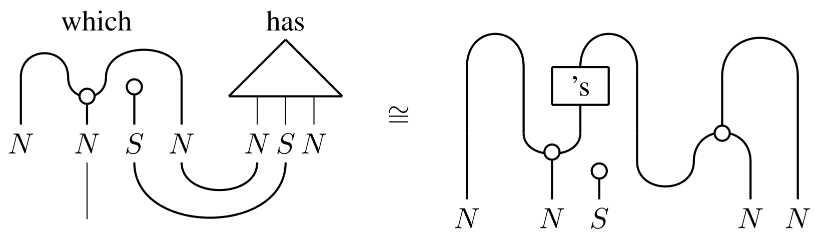

we do get the intended meaning:




Recently, we discovered that the functional word 'whose' as a relative pronoun can be decomposed in 'which' and 'has', via the map-state duality and dimensionality reduction on 'has', as shown below:



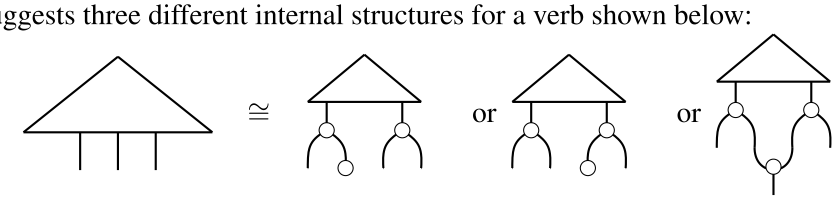

One is then able to derive the meaning of 'whose' from meanings of 'which' and 'has' as follows:




This demonstrates how from certain simpler functional words one can deduce the meaning of more complex ones. In this work package we aim to find the building blocks of functional words and then combine them to obtain other functional words. So far, the words that allow for unification of meanings across the sentence seem to be crucial building blocks. A word such as 'which' unifies the information of its head noun with the information of the rest of the sentence. For instance, in the clause 'dogs which eat meat', the word 'which' is unifying the meaning of 'dogs' with subjects of 'eat meat', in other words 'meat-eaters'.

**W2b. Equations for non-functional words**

Meanings of words can be assigned in terms of definitions taken from dictionaries, in terms of a smaller set of words. Possibly, also structural components may be used to 'construct' the meaning of words of compound types. This, for example, has already been investigated for the case of verbs, where Frobenius algebras allowed to reduce the dimensional requirements for representation of the verbs [13]. As shown below, a verb which is normally a 3-legged triangle and an element of the tensor space $N \otimes S \otimes N$, is being constructed from a two legged triangle in $N \otimes S$ and by padding the elements of the remaining dimension with 0. This procedure suggests three different internal structures for a verb shown below:




Experiment has shown that choosing one specific structure depends on the task, for instance, in disambiguation tasks where the object of the verb plays a crucial role, the middle case has performed best because it makes the role of the subject more explicit in the verb. Examples of such verbs are 'to file nails' versus 'to file books'.

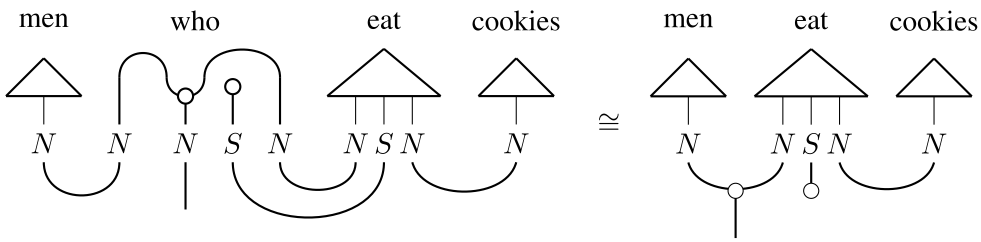
In this work package, we aim to extend the above methods and study the internal structure of words based on their denotations and their conceptual meaning. So far words such as 'John' and 'justice' and 'man' have all been treated uniformly and as nouns. Where as they each have a very different denotation and conceptual meaning. Proper nouns are very rare in a corpus and their occurrence may not mean much, unless they stand for famous characters. Concept nouns, however, may only acquire meaning as a result of their non-denotational properties, "justice" is one such example: it does not refer to any actual object in the real world. The word "man" on the other hand, is a common noun and can be seen as the sum of the individuals that are male. These considerations allow us to formalise the internal structures of words and use these when combining their meanings in building complex units such as sentences.

## W3. Automation

When working in distribution-only models one needs to do concrete calculations on real number vectors which have dimensions in the range of $2 \times 10^3$ to $6 \times 10^6$. The latter are all the unique words of one of the medium-sized copora we work with (the British National Corpus) and the former are the 2000 most occurring words of it. In compositional distributional models, one has to work with matrices and tensors of various ranks. Hence these dimensions will become exponentially larger. For instance an intransitive verb is an element of a space with tensor rank 2 (i.e. a matrix) and it will have dimensions, at least, $(2 \times 10^3) \times (2 \times 10^3)$. A transitive verb is an element of a space with tensor rank 3 (i.e. a cube) and will have dimensions, again at least, $(2 \times 10^3) \times (2 \times 10^3) \times (2 \times 10^3)$. Further, the various vectors, matrices, and cubes of words within a phrase and sentence need to interact with each other via matrix-multiplication and tensor-contraction operations to produce the meaning of the phrase and sentence. The corresponding computations will very quickly become hairy and intractable. The need for automations is greatly felt. Below, we suggest two solutions to this problem, they complement each other and can be used in parallel.

### W3a. Automating the meaning computations

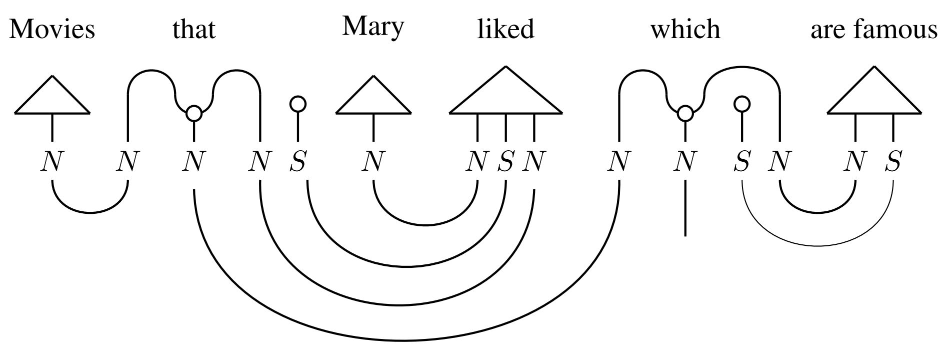
One of the tangible tools of the seminal work of the PI and Abramsky [1] has been a diagrammatic calculus, proven sound and complete with regard to the high level categorical language developed for quantum protocols. These diagrams abstract away the concrete computations performed on numbers and provide a high level view of the interactions that happen between the meanings of the words within a sentence, and among vectors, matrices, and tensor spaces in general. They have proven extremely useful in the language application and in simplifying the computations of the compositional distributional models. For example, to be able to derive the information-flows in a relative clause, one does not need to compute the corresponding vectors of the words therein to the end detail. All that is needed is to draw the corresponding meaning diagram, normalise it, and trace the strings. For instance, the meaning of a relative clause such as 'men who eat cookies' and its normalised form are as follows:




The vector meaning of the clause can then be read from the normalised diagram; for the above example this is as follows:

$$\overrightarrow{\text{men who eat cookies}} = \overrightarrow{\text{men}} \odot (\overline{\text{eat}} \times \overrightarrow{\text{cookies}})$$

Normalising the diagrams becomes harder as the sentences and phrases become longer and more complex. For instance, consider the following diagram which is not any more easy to normalise just by tracing the strings by eye:




Normalization can be automated by developing a software that does graph-reductions and then uses a computer algebra tool to represent the elementary shapes and strings to their linear algebraic forms. A similar tool has been developed in the PI's research group for normalising diagrams of quantum protocols in the software package *Quantomatic* [19]. In this work package, we aim to extend and tailor *Quantomatic* to language-related operations and reductions. A preliminary feasibility study has been performed by a former undergraduate project jointly supervised by the PI and CO-PI in Oxford and the new additions and alterations will follow the recommendations of that project.

### W3b. Quantum-computational speedup

The direct formal analogy between the compositional distributional model of meaning and the high-level semantics for quantum computing of [1, 6, 4], makes the idea that quantum computational speed-up may be highly beneficial for tasks proposed in this project. In particular quantum search, in the light of the importance for search in natural language processing tasks, deserves a serious analysis from our perspective. In particular, in recent work quantum search algorithms have been recast in the same diagrammatic language that we have employed above [20]. The idea of applying the speed-up of quantum algorithms to language tasks such as document search is a novel idea that has not been explored much. This is mainly due to a lack of a common language between the two fields of linguistics and physics. Our approach overcomes this difficulty. We aim to use our common diagrammatic and categorical languages to depict the general patterns of information flow in both fields, develop quantum-like algorithms for language search, and apply these to tasks for empirical validation and impact.

## W4. Empirical Validation

As the theoretical constructions and underpinnings are extended to cover larger and larger fragments of language, we will also need to provide empirical evidence for the new constructions. The initial information provided to us by distributional models vary from corpus to corpus and task to task. Experimentation will make sure the constructions can be carried over from one concrete setting to another and suggests parameters and domain specific adjustments. We work with corpora such as the British National Corpus (BNC), UKWac and Google's n-gram corpus. The sizes of these corpora vary from each other and their data is collected from difference sources. For instance, the BNC contains a snapshot of news on one specific day in early 90's. UKWac includes all the Wikipedia articles, and Google's n-grams corpus contains millions of books from the past couple of centries. For our purposes, the information in these corpora have to be parsed and each word has to be tagged with its corresponding grammatical relations. Automatic parsers that cover large-scale corpora are available, an example is $C\&C$, developed jointly by our close collaborator S. Clark (Cambridge). Then one has to build initial vectors and matrices and cubes depending on the grammatical role of each word. To this extent, for each experiment we repeat the above cycle in a different way, depending on the programmers involved. In this package, we aim to provide a uniform platform for such experimentation. Further, we will develop necessary datasets and gold standards to compare the performances of our model with human judgement and other models. As an example, consider the first few entries of a handcrafted dataset below:

| | Term | Description |
|---|---|---|
| 1 | emperor | person who rule empire |
| 2 | queen | woman who reign country |
| 3 | mammal | animal which give birth |
| 4 | plug | plastic which stop water |
| 5 | carnivore | animal who eat meat |
| 6 | vegetarian | person who prefer vegetable |
| 7 | doll | toy that girl prefer |
| 8 | football | game that boy like |
| 9 | skirt | garment that woman wear |
| 10 | widow | woman whose husband die |

This dataset was developed to experiment with vectors of relative pronouns on a term/description classification task. The goal of this task is to automatically assign the right descriptions to the words by computing the similarity between the term and the description. So far our model has outperformed other models, but for these results to be reliable and make an impact, we need to build the corresponding datasets from real corpora and the phrases and clauses seen in those corpora. This is the kind of task this project will help us achieve in large scale.

## References


[1] Samson Abramsky and Bob Coecke, 'A categorical semantics of quantum protocols', *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science*, 2004.

[2] Stephen Clark and Bob Coecke, Mehrnoosh Sadrzadeh, The Frobenius Anatomy of Relative Pronouns, 13th Meeting on Mathematics of Language (MoL), special interest group of ACL, 2013.

[3] Bob Coecke (2008) *Axiomatic description of mixed states from Selinger's CPM-construction*. Electronic Notes in Theoretical Computer Science **210**, 3–13.

[4] B. Coecke and R. Duncan (2008) *Interacting quantum observables*. In: Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP), pp. 298–310, Lecture Notes in Computer Science 5126, Springer-Verlag. Extended version (2009): *Interacting quantum observables: categorical algebra and diagrammatics*. arXiv:0906.4725

[5] Bob Coecke and Keye Martin. A partial order on classical and quantum states. In B. Coecke, editor, New Structures for Physics, Lecture Notes in Physics, pages 593683. Springer, 2011.

[6] Bob Coecke, Eric Paquette and Dusko Pavlovic, *Classical and quantum structuralism*. In: Semantic Techniques for Quantum Computation, I. Mackie & S. Gay (eds), p. 29–69, Cambridge University Press, 2009. arXiv:0904.1997

[7] Bob Coecke, Mehrnoosh Sadrzadeh and Stephen Clark, Mathematical Foundations for a Compositional Distributional Model of Meaning, Linguistic Analysis, a Festschrift for Jim Lambek, J. van Bentham and M. Moortgat (eds)

[8] C. Fellbaum (ed), *Wordnet: An Electronic Database*, MIT Press, 1998.

[9] J. R. Firth, 'A synopsis of linguistic theory 1930-1955', *Studies in Linguistic Analysis*, 1957.

[10] P. Gardenfors, *Conceptual Spaces: The Geometry of Thought*, Bradford Books MIT Press, 2000.

[11] J.-Y. Girard, *Linear logic*. Theoretical Computer Science **50**, 1–102, 1987.

[12] Edward Grefenstette and Mehrnoosh Sadrzadeh, Experimental Support for a Categorical Compositional Distributional Model of Meaning, ACL Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP), 2011.

[13] D. Kartsaklis, M. Sadrzadeh, S. Pulman and B. Coecke, Reasoning about Meaning in Natural Language with Compact Closed Categories and Frobenius Algebras. In: Logic & Algebraic Structures in Quantum Computing & Information, J. Chubb, A. Eskandarian and V. Harizanov, to appear.

[14] J. Lambek, *From Word to Sentence*, Polimetrica, Milan, 2008.

[15] Keye Martin. Domain theory and measurement. New Structures for Physics, pages 491591, 2011.

[16] R. Montague, 'English as a formal language', *Formal Philosophy*, 189-223, 1974.

[17] H. Schuetze, 'Automatic Word Sense Discrimination', *Computational Linguistics* **24**, 97-123, 1998.

[18] Peter Selinger (2007) *Dagger compact closed categories and completely positive maps*. Electronic Notes in Theoretical Computer Science **170**, 139–163.

[19] Aleks Kissinger, Lucas Dixon, Ross Duncan, https://sites.google.com/site/quantomatic/publications.

[20] Jamie Vicary, The Topology of Quantum Algorithms. arXiv:1209.3917