A Cognition-Based Model of Text Meaning

Distributional Models of Meaning: Hilary Term 2019

Nicolas Köcher

March 27, 2019

ABSTRACT. We present a mathematical framework to model the meaning of a text, which is based on both Coecke's DisCoCirc model and Gärdenfors' theory of conceptual spaces. This combination lets us deal with some previously intractable elements, most notably allowing us to resolve the meaning of pronouns. We also give an outlook on possible tasks for further research.

Contents

1 Introduction

"There are dictionaries for words, so why aren't there any dictionaries for sentences?"

Bob Coecke [4]

This is the question that initially motivated the creation of *categorical compositional distributional models of meaning*, DisCoCat for short. In the seminal paper [2], Coecke combines two different theories: a quantitative approach that empirically generates a meaning vector for each word from a large text corpus, and the use of a pregroup to model the grammatical structure of a sentence. This structure can be reduced to a single element by the rules of the pregroup. By performing analogous "reduction" operations on the tensor product of the vectors that represent each word, it is in fact possible to calculate a single meaning vector for the whole sentence.

An alternative approach that has arisen only recently uses Gärdenfors' framework of *conceptual spaces* [7] in place of a distributional model. Originating from cognitive science, this theory is based on the way the human brain understands concepts as combinations of certain properties, such as size, colour and taste. Representing each noun in this way, we can once again obtain a meaning for the whole sentence by using a categorical compositional approach. As it is directly modelled on human cognition, the results of this framework are generally more tangible than with DisCoCat, where the meaning vectors are mostly only useful for sentence comparison. While there is currently no way to automatically generate a conceptual space for each word, Tyrrell [9] suggests that manual input is unavoidable, and in fact important as part of a learning process.

Both of these frameworks, however, only focus on single sentences. In [4], a first attempt was made to extend DisCoCat to whole texts in the form of DisCoCirc (*circuit-shaped* compositional distributional) models of meaning. The goal of this theory is to investigate how the meaning of words changes throughout a text, and what influence the order of sentences has. While the general ideas introduced in [4] seem to be very fruitful, combining them with the distributional framework leads to a few issues and limitations. Rather, the author believes that the paper's general method of analyzing texts naturally lends itself to an adaptation in a cognition-based model, using conceptual spaces as the base. Not only does this result in a clearly tangible output, but it also allows us to resolve some kinds of ambiguities more easily, such as those arising from pronouns.

After a brief recapitulation of the current state of research in Chapter 2, we introduce our framework in Chapter 3. Aside from providing several examples that showcase its features or justify our design choices, we also talk about possible directions for further expansion of the model. Chapter 4 deals with text comparison, before we finish the paper by analyzing a small story with the methods of our framework.

2 Compositional Models of Meaning

2.1 DisCoCat

In order to deduce the meaning of a sentence from those of the isolated words, it is necessary to understand the underlying grammatical structure. We solve this by assigning each word an element of a pregroup P as its type. The combined type for the whole sentence is then reduced to the sentence type s by using the pregroup axioms $x^l \cdot x \leq 1$, $x \cdot x^r \leq 1$; an in-depth explanation can be found in [2].

We can now choose any compact closed strict monoidal category $\mathbf{C}$ as our category of *meaning spaces.* Each type is associated with an object of $\mathbf{C}$, with adjoints being mapped to adjoints and pregroup products to tensor products. The morphisms $\epsilon^l : A^l \otimes A \to I$ and $\epsilon^r : A \otimes A^r \to I$ that are given by the compact closed structure of $\mathbf{C}$ now correspond to the pregroup reduction rules. Forming the tensor product of the meaning spaces of each word, we can therefore compose ϵ^l, ϵ^r and identities in an analogous manner to the type reduction and obtain a morphism from this tensor product to the *sentence space.* This morphism makes it possible to calculate the meaning of a sentence from those of the words that make it up.

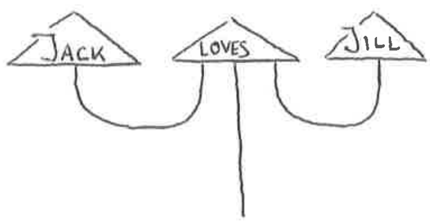
Example 2.1. *Consider a sentence with a simple* noun – transitive verb – noun *structure, such as "Jack loves Jill". We assign the noun type n to both Jack and Jill, while "loves" as a transitive verb gets the type $n^r s n^l$ (this is elaborated on a bit more in [9]). The sentence as a whole now has the type $n(n^r s n^l)n$. This reduces to s by*

$$n(n^r s n^l)n = (nn^r)s(n^l n) \leq 1 \cdot s \cdot 1 = s. \tag{1}$$

In $\mathbf{C}$, we identify the noun and sentence types n and s with the meaning spaces N and S, respectively. The whole sentence therefore lives in the space $N \otimes (N^r \otimes S \otimes N^l) \otimes N$. This can be reduced in the same way:

$$N \otimes N^r \otimes S \otimes N^l \otimes N \xrightarrow{\epsilon^r \otimes 1_S \otimes \epsilon^l} I \otimes S \otimes I = S. \tag{2}$$

Both the pregroup and the monoidal category lend themselves to a graphical representation via string diagrams, which allows us to depict Example 2.1 in a simple way:



Despite all efforts to change this convention, linguists sadly still prefer a top-down notation, which is contrary to the usual bottom-up orientation in quantum theory.

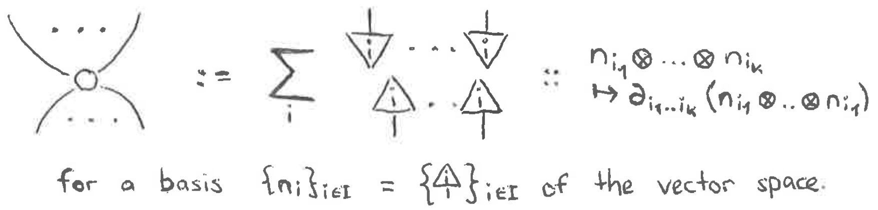
The conventional approach used for DisCoCat chooses **C** to be **FVect**. Meaning vectors are usually generated through quantitative analysis over a large corpus of text; [9], for example, chooses the five most common words as the basis for the noun space, and the coefficients correspond to the number of times a word appears in close proximity to the respective basis word. Other types of words are quantified in a similar way. For transitive verbs in particular, DisCoCat uses [6]'s suggestion of identifying the sentence space S with $N \otimes N$, representing verbs as sums of the form $\sum_{ij} c_{ij}(n_i \otimes (n_i \otimes n_j) \otimes n_j)$, with $\{n_i\}$ being the base of N and the constants c_{ij} generated through learning.

2.2 Internal wirings

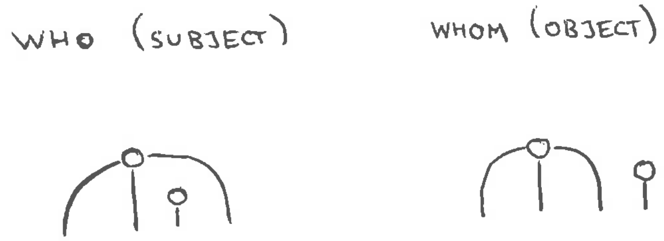
Linguistics differentiates between *content words*, which we considered in the previous section, and *functional words* such as "and" or relative pronouns. The meanings of the latter should obviously not be learned through a quantitative approach, but pre-defined as part of the framework. If there is no such definition, the relevant text corpus needs to be edited in order to replace these words, as is done with pronouns in [9]. Being able to handle as many functional words as possible, or even full unedited texts, is one of the long-term goals for successful models of meaning.

Depending on the purpose of the specific model, other words may also be represented in a particular pre-determined way; we will see an example by the end of this chapter.

One of the easiest functional words to model is "does" of type $n^r s s^l n$, as in "Jack *does* love Jill". This can be represented by a double cup (η-morphism) in the graphical calculus, which disappears upon contracting the diagram by the use of the yanking equations. A perhaps more interesting example are relative pronouns, whose structure was first explored in [3]. They can be represented using a familiar feature of string diagrams, so-called *spiders* or *multi-wires*:

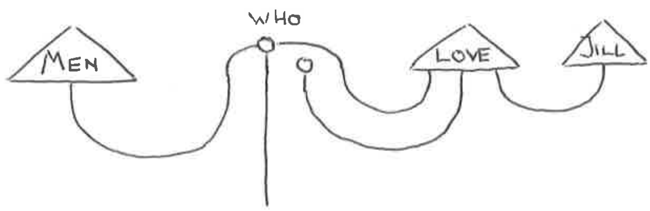


The two types, *subject* ("Men *who* love Jill") and *object* ("Women *whom* Jack loves") relative pronouns, have the types $n^r n s^l n^l$ and $n^r n n^l s^l$, respectively. Using spiders, we can represent their meanings as follows:



We can justify this notation by looking at the previous example "Men who love Jill".

Example 2.2. *Let N be spanned by $W \cup M$, where $W := \{n_1, .., n_w\}$ corresponds to the set of women and $\{n_{w+1}, .., n_m\}$ to that of men. In particular, n_1 represents Jill.*



If we define "loves" as $\sum_{ij} \ell_{ij}(n_i \otimes (n_i \otimes n_j) \otimes n_j)$ in accordance with the previous section, we can model the entire sentence as:

$$
\begin{aligned}
&(1_N \otimes \epsilon_S) \circ (\epsilon_N \otimes 1_N \otimes 1_S \otimes \epsilon_N \otimes 1_S \otimes \epsilon_N) \circ \Big(\Big(\sum_{i=w+1}^{m} n_i\Big) \otimes \Big(\sum_{j,k,l=1}^{m} n_j \otimes n_j \\
&\otimes (n_k \otimes n_l) \otimes n_j\Big) \otimes \Big(\sum_{p,q=1}^{w} \ell_{pq}(n_p \otimes (n_p \otimes n_q) \otimes n_q)\Big) \otimes n_1\Big) \\
&= (1_N \otimes \epsilon_S) \circ \Big(\sum_{i=w+1}^{m} \sum_{j,k,l,p,q=1}^{m} \delta_{ij}\delta_{jp}\delta_{q1}\ell_{pq}(n_j \otimes (n_k \otimes n_l) \otimes (n_p \otimes n_q))\Big) \qquad (3) \\
&= (1_N \otimes \epsilon_S) \circ \Big(\sum_{i=w+1}^{m} \sum_{k,l=1}^{m} \ell_{i1}(n_i \otimes (n_k \otimes n_l) \otimes (n_i \otimes n_1))\Big) \\
&= \sum_{i=w+1}^{m} \sum_{k,l=1}^{m} \delta_{ki}\delta_{l1}\ell_{i1}n_i = \sum_{i=w+1}^{m} \ell_{i1}n_i.
\end{aligned}
$$

This is a mixture of all male base vectors, weighted by their love to Jill. If we consider love to be a binary relation, we can indeed recover the definition.

In the same way that spiders provide an intersection between the properties "is a man" and "loves Jill", they can also be used to model certain types of adjectives, as shown in [1]. So-called *intersective adjectives*, such as those that denote colour, gender or other properties that clearly specify a subset of the noun space, can be modelled by "attaching" that subset to the noun with a three-way spider. Other adjectives however vary in meaning depending on the noun they are attached to, such as "soft", which has a different softness threshold for clay and for bananas; some, like "dead", even create new meanings that the noun by itself does not include. These two kinds of adjectives can generally not be represented by an internal structure.

2.3 Conceptual spaces

[1] first introduced the concept of defining nouns in terms of human cognition rather than empirically calculating their meaning. The noun space in that model is the Cartesian product of finite intervals, corresponding to different attributes by which the brain classifies concepts; rather than as a vector, we represent a noun as a convex subset of that space. A more detailed explanation, along with comprehensive examples, can be found in that paper.

For this model, we choose **ConvexRel** as our category of meaning spaces, in which the morphisms are convexity-preserving relations. Like **FVect**, this category allows for ϵ and η morphisms, which are the usual caps and cups from **Rel**, as well as spiders. As each subset of the noun space can be seen as the union of its single points, just like a vector is a linear combination of its base vectors, a spider $N \otimes \cdots \otimes N \to N \otimes \cdots \otimes N$ is therefore defined by $(x, .., x) \mapsto \{(x, .., x)\}$ and $(.., x, .., y, ..) \mapsto \varnothing$ for any points $x \neq y$ in that space. It is evident from this definition that a spider morphism, in fact, returns the intersection of multiple sets. As such, it can be used to represent relative pronouns and intersective adjectives in a much more natural way than the vector-based model.

One can indeed think of a convex subset as a direct representation of the knowledge we have about a noun. The more we learn about a subject, the more does this knowledge restrict the corresponding set. For example, in a model that includes a colour property in the form of a three-dimensional RGB space, applying the adjective "red" to a word intersects its colour dimension with the subset of colours that are perceived as red.

Verbs, however, pose a problem in this model, as there is no obvious choice for a sentence space. Identifying S with $N \otimes N$ hardly makes as much sense as in the vector-based model, particularly for sentences that do not have exactly one subject and one object. [1] offers various examples, each of which only provides a specialized model focusing on one specific aspect of meaning for a pre-defined type of sentence. While this shows the versatility of the conceptual framework, it clearly lacks the elegance of DisCoCat's ability to compare any two sentences regardless of grammatical structure.

2.4 DisCoCirc

A recent development is the idea that rather than unifying sentence types and comparing single sentences, we should instead switch to analyzing whole texts. If we fix a sentence type S, this is only possible by connecting all of them with spiders, which [4] calls the *bag of sentences* model. The reason for this is that since spiders fuse with each other, modelling a text in this way would remove any effects of sentence order. It can however be argued that ignoring the order of sentences has non-neglegible consequences (as Coecke illustrates by the example of a recipe).

In order to solve this problem, and simultaneously investigate the extent of the effect that sentence order imposes upon a text, [4] introduces a model that omits the existence of a sentence type altogether. DisCoCirc, which serves as a *circuit-shaped* generalization of DisCoCat, instead assigns each actor that appears in a text its individual type. In place of a sentence space, the output of a verb is a combination of the meaning spaces for these actors. By arranging the whole text in a graphical circuit with the actor wires running from top to bottom, it is even possible to see temporal order and causal relations.

This idea in itself is very versatile and allows the DisCoCirc framework to be used for many specialized purposes. To mention a simplistic example that appears in [4]: by identifying the verb “knows” with a 4-way spider, the whole diagram for a text consisting of sentences of the form “A knows B” contracts to a (transitive) knowledge graph.

3 A Cognition-Based Model of Text Meaning

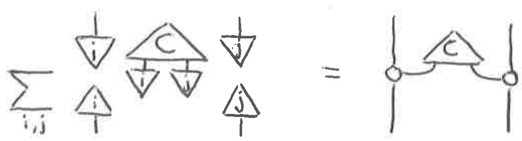
In this chapter, we will present a cognition-based framework for understanding text meanings that operates in **ConvexRel**, using the DisCoCirc notion of actor wires. We shall first justify our use of conceptual meaning spaces over vectors, before outlining the basic functionality of our model and the ways it resolves certain aspects of language.

3.1 Why conceptual spaces?

As a distributional model of sentence meaning was the base for the DisCoCirc approach, one may ask why that should not be the preferred framework. Indeed, compared to DisCoCat, which has been prevalent for almost a decade, the use of Gärdenfors’ model is a very young approach. Still, the choice of **FVect** as our category of meaning spaces has several drawbacks, while we can remedy these to an extent and reap further benefits when choosing a cognition-based framework.

The most obvious issue with vector models is that of choosing a basis for the noun space. Usually, the basis vectors are chosen to be the most common words in a text corpus, or even identical to the actors. [9], for example, includes Anakin and Palpatine as elements of his basis, who are simultaneously the two main characters of the text corpus. Similarly, in Example 2.2, which concerned itself with relations between people, we chose the basis to be the set of all men and women. These choices do not make much sense if each actor is represented by their own wire type. While we can certainly choose the basis to be a set of attributes instead, a vectorial framework would force every word to be linear in any of these properties. As a possible side effect of this, two actors who are both humans might for instance share very similar coefficients for most basis vectors of the noun space, making it hard to accurately model verbs that have very different effects for the two of them. **ConvexRel** on the other hand grants a lot more flexibility in choosing a sensible basis, as a noun can be described by a convex subset of the meaning space rather than a single point, and the concept of linearity does not exist.

A second problem arises from the way verbs are modelled in DisCoCat. As noted in Section 2.1, we represent a seemingly 4-dimensional verb, which lies in the space $N \otimes S \otimes N$ (or in the case of DisCoCirc, $N \otimes (N \otimes N) \otimes N$), as $\sum_{ij} c_{ij}(n_i \otimes (n_i \otimes n_j) \otimes n_j)$ with c_{ij} being entries in a (2-dimensional) $\dim(N) \times \dim(N)$ matrix C. If we choose the top-down depiction and write the verb as an endomorphism on $N \otimes N$, this becomes $n_i \otimes n_j \mapsto c_{ij}(n_i \otimes n_j)$. Graphically, we can simplify this to:



From this notation, it is clear that if we use the vector-based model for text meanings, each text contracts to a bag of sentences, defeating the purpose of our framework.

Furthermore, DisCoCirc has no clear notion of information gain. At any point each actor is only represented as a linear combination of arbitrary base vectors, and due to the aforementioned problems in choosing a base, purity or mixedness does not make sense as a measure of informational content. This could arguably also create an issue with intersective words, as the properties they denote might not be preserved throughout the following sentences in this model. A conceptual framework on the other hand has the advantage of representing actors as subsets, which gives both a clear notion of intersection and the possibility of measuring information, for example by defining a volume measure or simply by the number of dimensions of a set.

One additional advantage of our chosen model is the tangibility of its output. While DisCoCat has proven to be very effective in comparing sentences, a human reader will not understand the meaning of an isolated vector outside of very trivial cases. This problem is obviously avoided by modelling noun spaces on human cognition.

3.2 A simple model

In this section, we will be giving a basic outline of a possible framework along with a relatively simple example. While some of the model's intricacies will be explained later in the chapter, we leave the detailed specifications open for future research.

The basis of our model is given by the noun space, which we intentionally do not specify in too much depth. In order to be capable of modelling both human and inanimate actors accurately, this space needs a lot of dimensions that may not all be compatible with each noun. For example, if we go by Tyrrell's model from [9], a "human propensity" dimension does not make much sense for an apple, neither do colour, taste and texture for a human. Regardless, all these classifications exist in the human mind, even though we only pay attention to some of them for each noun. We shall therefore assume that if a particular attribute is not given, the respective noun spans the whole possible space in this dimension.

For the purposes of this paper, we will only consider a relevant subset of the possible attributes a complete model should include. These are: size, age, mood (each on the interval $[0, 1]$, which [9] makes a more detailed case for), colour (as a 3-dimensional RGB cube), location and gender. Location is generally hard to model and tends to include references to other objects, which was explored more in [8] and is an entirely different issue for models of text meaning; however, for the scope of this project, a simplification to $[0, 1]$ shows to be sufficient. We also represent gender on a 1-dimensional interval of $[-1, 1]$, with 0 for male, 1 for female and -1 for neutral. This lets us resolve gendered pronouns, and we can initialize the gender as -1 for inanimate nouns and $[0, 1]$ for actors with a name. Our whole 8-dimensional meaning space is therefore:

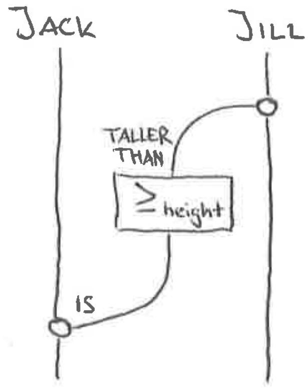
$$N := N_{size} \times N_{age} \times N_{mood} \times N_{colour} \times N_{location} \times N_{gender}. \tag{4}$$

As in [4], we choose not to define a sentence space S. In place of that, each verb outputs as many wires of type N as there are subjects and objects. Each actor therefore has one wire representing them that runs through the whole diagram, allowing sentences to compose sequentially by connecting the output of one with the corresponding input of another. To enable a cleaner notation that lets all actor wires run straight from the top to the bottom of the diagram, we choose a different representation for words in the category: rather than modelling all words as states, we make them into morphisms from an input space to an output space. Inputs can generally be distinguished from outputs by the fact that they are left or right adjoints in the pregroup model. For instance, a transitive verb of type $n^r s n^l$ will no longer be written as a state $I \to N \otimes (N \otimes N) \otimes N$, but as a morphism $N \otimes N \to N \otimes N$ (Subject $\otimes$ Object $\to$ Subject $\otimes$ Object).

For every morphism we construct, we make sure that every subject or object wire at the top has exactly one corresponding wire at the bottom, with the possible exception of the word "is" (in the form of a two-to-one spider [4]), which may identify two actors with each other. While all existing actor wires are therefore preserved, every new subject or object on the other hand creates an additional wire due to the possibility of it being referenced later. This may result in an abundance of unnecessary wires; these can, however, be terminated and ignored when it comes to text comparison.

The advantages of our framework are showcased well by a single example:

Example 3.1. *Consider the sentence "Jack is taller than Jill". For the sake of clarity and simplicity, we model "taller than" as a single word of type* nn^l*. As this corresponds to a morphism* $Jill \to N$ *returning the set of points with a greater height than Jill, we need to copy Jill's state first in order to preserve the wire, resulting in a composite morphism* $Jill \to N \otimes Jill$*. This set is then intersected with Jack's meaning subset by "is"* : $Jack \otimes N \to Jack$*. The entire sentence can therefore be modelled as:*



Note that "Jill is shorter than Jack" has an identical meaning, as is evident from the symmetrical structure of the diagram, along with "shorter than" and "taller than" being the relational converse of each other.

Let us compute what the composition of these morphisms actually does: consider two single points $k \in Jack$, $l \in Jill$*. One may think of them as possible configurations for Jack and Jill, given the current knowledge we have about them. Writing* $k = (size(k), \hat{k})$, $l = (size(l), \hat{l})$ *with* $\hat{k}, \hat{l} \in N_{age} \times \cdots \times N_{gender}$*, we have:*

$$\begin{aligned} &(Jack\ is\ taller\ than\ Jill)(k \otimes l) \\ &= (is \otimes 1_{Jill})(1_{Jack} \otimes taller\ than \otimes 1_{Jill})(1_{Jack} \otimes copy)(k \otimes l) \\ &= (is \otimes 1_{Jill})(1_{Jack} \otimes taller\ than \otimes 1_{Jill})(k \otimes l \otimes l) \\ &= (is \otimes 1_{Jill})(k \otimes \{n \in N; size(n) \geq size(l)\} \otimes l) \\ &= (\{k\} \cap \{n \in N; size(n) \geq size(l)\}) \otimes l \\ &= \left\{ \begin{array}{ll} k \otimes l, & size(k) \geq size(l) \\ \varnothing, & \text{otherwise.} \end{array} \right\} \end{aligned} \tag{5}$$

The sentence therefore filters out all pairs of points where Jack is not taller than Jill. The individual heights of both Jack and Jill are not restricted, however, they have to *obey this relation and are therefore interdependent. In a vector space model that works with linear maps instead, encoding such a relation would not be possible.*

3.3 Resolving pronouns

As our framework is so heavily based on intersection, it offers a surprisingly effective way to resolve the meaning of pronouns. The vast majority of pronouns that appear in texts refer to an actor that appears in the directly preceding sentence, either as a subject or an object. As opposed to other languages like Spanish, English also has the advantage that pronouns can never be omitted, and therefore always determines at least the gender, if not the identity of an actor. These two properties – appearing as an actor in the previous sentence, and gender – are in fact the only indicators for the meaning of a pronoun. We shall prove this by means of a few simple examples.

Example 3.2. *Alex reads a book. Meanwhile, Jordan watches TV. She is tired.*

While the pronoun could theoretically refer to both Alex and Jordan (note that these are gender-neutral names), it is clear to the reader that we are talking about Jordan, showing that appearing in the previous sentence is in fact important.

Example 3.3. *Alex lives with his wife Jordan. He is coming home from work.*

Both Alex and Jordan are actors in the previous sentence, but since a gender has been given for both, the pronoun "he" can be unambiguously resolved. Note that the possessive pronoun in the first sentence refers to Alex, who appears in the same sentence. This is an exception, as any other pronoun would turn into "him-/herself" here.

Example 3.4. *Alex thinks of Jack. He had just been talking to him.*

While there is some bias towards subject and object retaining their roles, it is often not entirely clear, as we can see from this example. Both Jack and Alex could be the subject of the second sentence, creating sufficient ambiguity to reflect this in the model.

Example 3.5. *Jack works at night. Alex has a few drinks with his wife.*

This example showcases how the possessive pronoun may either refer to a previous actor (Jack) or take a reflexive role. Let's hope for Jack's sake that the latter is the case.

From our observations, we can now deduce a concrete method for resolving pronouns. While there is no direct way to rewrite them within the diagrammatic calculus, we can instead model them as the union of several string diagrams, which represents ambiguity and is a direct analogon to the sum in **FVect**. One may think of this as a kind of overlay.

Algorithm 3.6. *For a sentence containing non-reflexive pronouns, consider any diagram fulfilling the following criteria:*

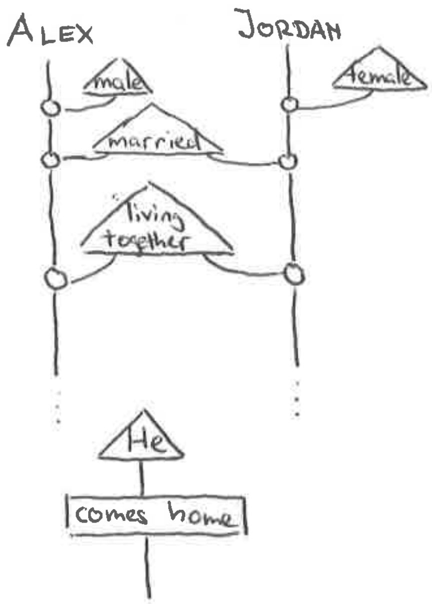
- *In place of any personal pronoun, that wire is connected to any distinct actor wire from the last sentence (such that no two pronouns represent the same actor), and the corresponding gender adjective (male/female/neutral: specifying subset* $\{n \in N; gender(n) = 0, 1, -1\}$*) is attached to the wire.*

- *In place of any possessive pronoun, that wire is either connected to any distinct actor wire from the last sentence, attaching the corresponding gender adjective, or the pronoun is marked as reflexive.*

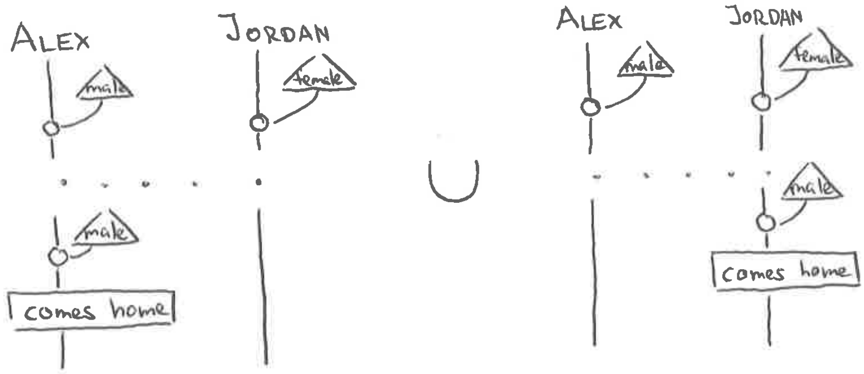
The union of all resulting diagrams is now an equivalent representation of that sentence.

We can directly see the usefulness of Algorithm 3.6 from the following example:

Example 3.7. *Consider the two sentences from Example 3.3. If the pronoun is unresolved, the diagram looks like this (ignoring details of little relevance):*



The pronoun can now stand for either Alex or Jordan, and the text therefore represents the union of these two possibilities:

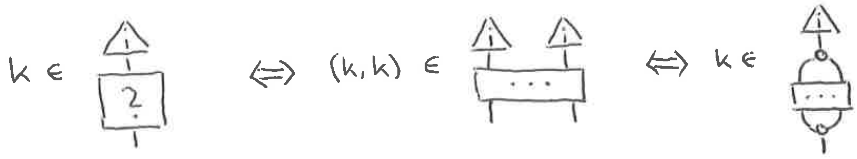


However, the state on the right is empty, as Jordan's wire is first intersected with "female" (by defining her as Alex' wife), then with "male" (by assuming the pronoun applies to her). These two adjectives are conflicting, therefore we can ignore the diagram on the right and connect the pronoun's wire to Alex with certainty.

3.4 Reflexivity

Reflexive pronouns remain an issue of our framework. While they can always be assigned without ambiguity, the difficulty lies in defining how an actor can simultaneously act as subject and object of the same sentence. As we will see, this is not always entirely possible within our model.

Consider a transitive verb, modelled as a morphism $N \otimes N \to N \otimes N$, and let the object of this verb be a reflexive pronoun. The verb therefore only needs one actor as its input, and as the subject and object should still be the same afterwards, it only needs to output one noun either. It remains to find a way to construct a morphism $N \to N$ that corresponds to the reflexive application of this verb. As the following consideration shows, we obtain this morphism from applying spiders at the top and bottom of the original box:



Let i be an element of N, and we consider any possible combination k of attributes that an actor with previous state $\{i\}$ may take after a reflexive application of this verb. This combination can occur if it is simultaneously possible for subject and object, after both are initialized with i. We can rewrite this using two spiders, as single elements are exactly the copyable states of our framework. Obviously, depending on the exact pronoun, the corresponding gender attribute should also be added to the wire.

Note that this method of resolution is not perfect: We can for instance consider the sentence "[Subject] makes [Object] happy", which can arguably be modelled by the mood of the object being set to (not intersected with!) $[0.8, 1]$, while nothing happens to the subject. However, we run into a problem when trying to resolve "Jack is sad. Jack makes himself happy". As the subject, Jack's mood does not change, and when a happy Jack is intersected with a sad Jack, the text contradicts itself. It is implied that the mood change of the object automatically applies to the (identical) subject too; however, a compositional model does not offer any easy solution to this.

3.5 Going beyond

The model presented in this chapter is merely an outline of a complete framework, and while it can deal with several issues that were previously considered intractable, we are still a long way from being able to understand the meaning of every text. This final section aims to propose a few ideas that our model could potentially incorporate.

One useful notion might be the introduction of an "ambient" wire to deal with sentences that have no subjects or objects, and as such, do not influence any actors. A simplistic example for such a sentence is "It rains", where the verb "rains" would be modelled by a morphism Ambient $\rightarrow$ Ambient. This choice of typing would also unambiguously determine "it" not to be a pronoun, but the absence of an actual subject. Other actors might potentially interact with the ambient state through certain verbs, although the meaning space would need to be drastically expanded for a sentence such as "It rains" to make sense in our model.

An important issue has always been posed by the word "and". If it connects two sentences, we can simply treat them as separate, however both should be considered for pronouns in the next sentence. If it connects two actors, though, the meaning may vary, specifying interaction or simply parallel action depending on the context and possibly even creating ambiguity between both ("Alex and Jack are married"). Resolving this may be well beyond the scope of the present work, even though it is certainly an attractive task for future research.

While we previously touched upon possessive pronouns, it might be difficult or outright impossible to represent relations between actors, such as ownership, within the bounds of the model. One can simplify them as semi-cartesian verbs [5], but no actor can directly refer to another. A partial solution to this might lie outside of the model, by marking an actor wire as belonging to another ("Jack", "Jack's hat"). Still, the representation of symmetrical relations, such as love, remains an open problem.

One last question that was already brought up in [1] is the notion of negation, as the complement of a convex set is generally not convex. However, the author believes that such negated properties are never an important part of a concept, rather serving as additional information on top of our conceptual space, in a similar way to ownership. Therefore we should not need to represent them within our framework.

4 Comparing Text Meanings

By means of the vectorial scalar product, the distributional framework offers a simple way to calculate a degree of similarity between sentences. This can be extended to entire texts, provided they have an identical number of actors, which can easily be reached by discarding all unnecessary wires. **ConvexRel** on the other hand has no such product, and directly connecting two sentences or texts directly only yields a Boolean value. Nevertheless, we can still define some ways to calculate text similarity.

Let us consider a general noun space of dimension m, as well as two texts with n relevant actor wires each, which are in a one-to-one correspondence. Each text therefore specifies a convex subset of $N^{\otimes n} \subseteq \mathbb{R}^{n \cdot m}$ as its meaning. A first, primitive way of comparing these is by simply connecting corresponding wires with n cap morphisms, yielding a composite of type $I \to I$. This is an analogon to the vectorial scalar product and returns $true := \{(*, *)\}$ exactly if these two subsets have nonempty intersection. Through this method of "comparison", we find out if two texts are at all compatible.

A somewhat more nuanced way to compare two texts arises by looking at their intersection and, assuming it is non-empty, measuring its dimension which represents the degrees of freedom or ambiguity. We can obtain this intersection by combining each pair of corresponding actor wires with a two-to-one spider. Comparing its dimension to those of the original sets can, in a way, measure similarity.

Example 4.1. *As an oversimplified example, consider two texts in our 8-dimensional noun space, talking about a single inanimate actor. One describes this actor to be red, specifying the 4-dimensional subset $N_{size} \times N_{age} \times N_{mood} \times (1, 0, 0) \times N_{location} \times \{-1\}$, while the other gives an exact size, age and location, leaving the 4 dimensions of mood and colour unclear. Their intersection, however, fixes every attribute aside from mood and is therefore 1-dimensional. So while these two texts might be talking about the same red car, they are very dissimilar in what they do and do not specify.*

One caveat of this approach is the fact that we rarely obtain definite values for any attribute of our model, with the obvious exception of gender. Realistically, "red" specifies the three-dimensional subspace $[0.8, 1] \times [0, 0.2] \times [0, 0.2]$ rather than the exact colour value $(1, 0, 0)$; and even a statement such as "Jack is the same height as Jill", which one might think would reduce the dimension by 1, should optimally be modelled with a few millimeters of tolerance.

The case that both sentences and their intersection have the same dimension is therefore more common than one might assume. For this case, we can define a measure on that subspace, with the obvious choice being the (Lebesgue) volume measure. By dividing the volume of the intersection by that of the larger sentence subset, we obtain a coefficient for similarity. However, this measure is easily skewed by varying tolerance parameters, as the seemingly minor difference between $[0.98, 1]$ and $[0.99, 1]$ will result in a halved similarity coefficient when compared to $[0.9, 1]$.

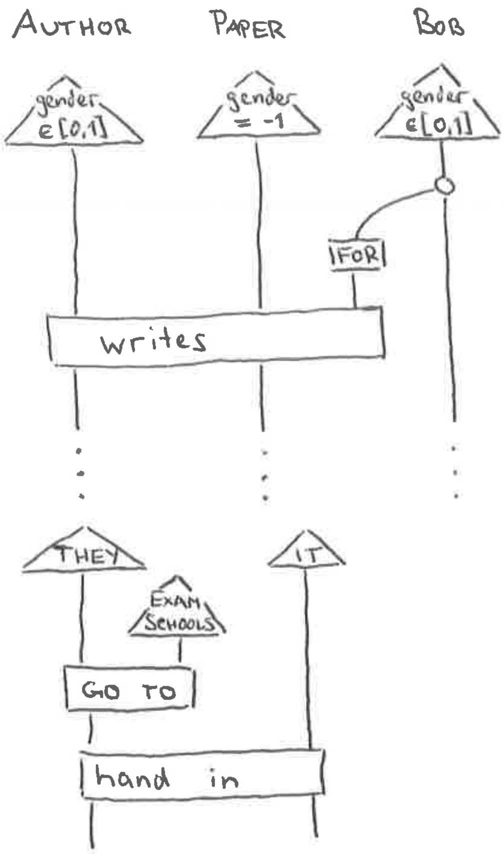
An entirely different approach, albeit a more tedious one, would be to determine a measurement for distance rather than similarity of two concepts. Using a probability distribution over $N^{\otimes n}$ that is uniform with regards to the aforementioned volume measure, one could for instance calculate the average Euclidean distance between two points in the two sets. This idea poses a few problems though: not only does $[0, 1]^{nm}$ have a greater distance coefficient to itself than to $\{0.5\}^{nm}$ under this model, but it might also be troublesome to define a notion of Euclidean distance for non-hyperrectangular spaces in the first place. While a volume measure is independent of distortion, distance in the taste tetrahedron, for instance, is directly affected by our choice of embedding into $\mathbb{R}^3$. Still, at least for $N \simeq [0, 1]^m$, the minimum distance could be a way to compare two sets without a common element, and the average minimum distance of points in one set to the other set might be a reasonable choice for a general distance coefficient.

5 A Story

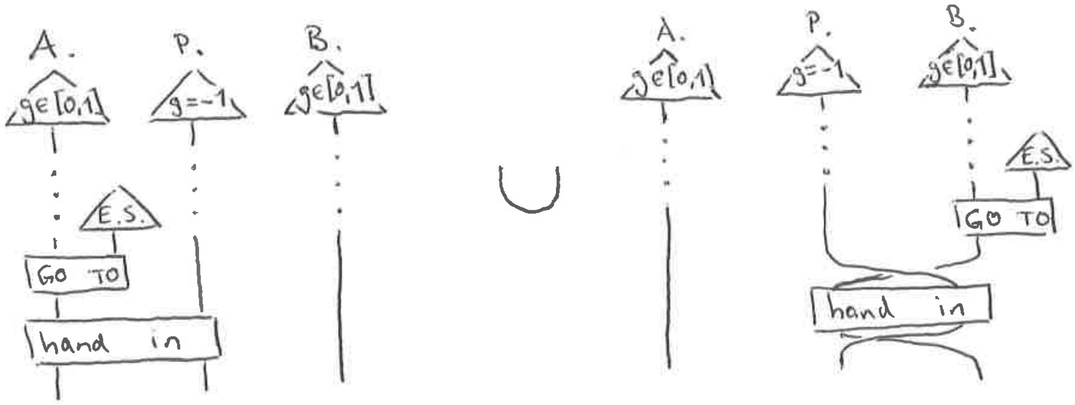
> "The author writes a paper for Bob. They go to the Exam Schools and hand it in. Bob goes from the CS building to the Exam Schools. He likes the paper. The author is happy."

Some things to note: As the meanings of "Exam Schools" and "CS building" do not change throughout the text, we model both as static rather than dynamic nouns for the sake of simplicity. We can, however, assume that their location attributes are disjoint. The verbs "write", "hand in" and "like" can obviously not be represented all too accurately by our limited model; obviously none of them changes the gender of either actor though, and neither the location of the subject. "They" is meant as a neutral pronoun, since at the time when this paper was originally written, the exam conventions prevented the author from clarifying his gender.

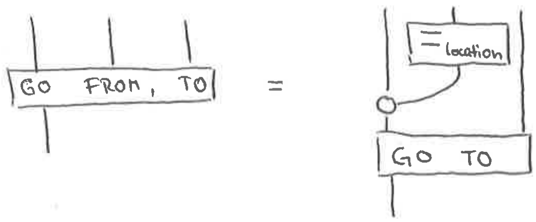
The first two sentences can be modelled as follows:



We can now resolve the two pronouns in six different ways. However, if we connect "it" to the author or Bob, we get a mismatch, as these have already been initialized with a different gender. Therefore, "it" has to be connected to the paper, and the whole diagram becomes equivalent to the union of these two (non-empty) states:



We model the verb “go to” to set the location of the subject to that of the object. Usually, the object would need to be copied first, but as the Exam Schools are a static noun for the purpose of our story, there is no need for that. Similarly, we can define an additional parameter “from” for “go to”, which works as an assertion that determines the subject’s previous location, filtering out impossible cases:



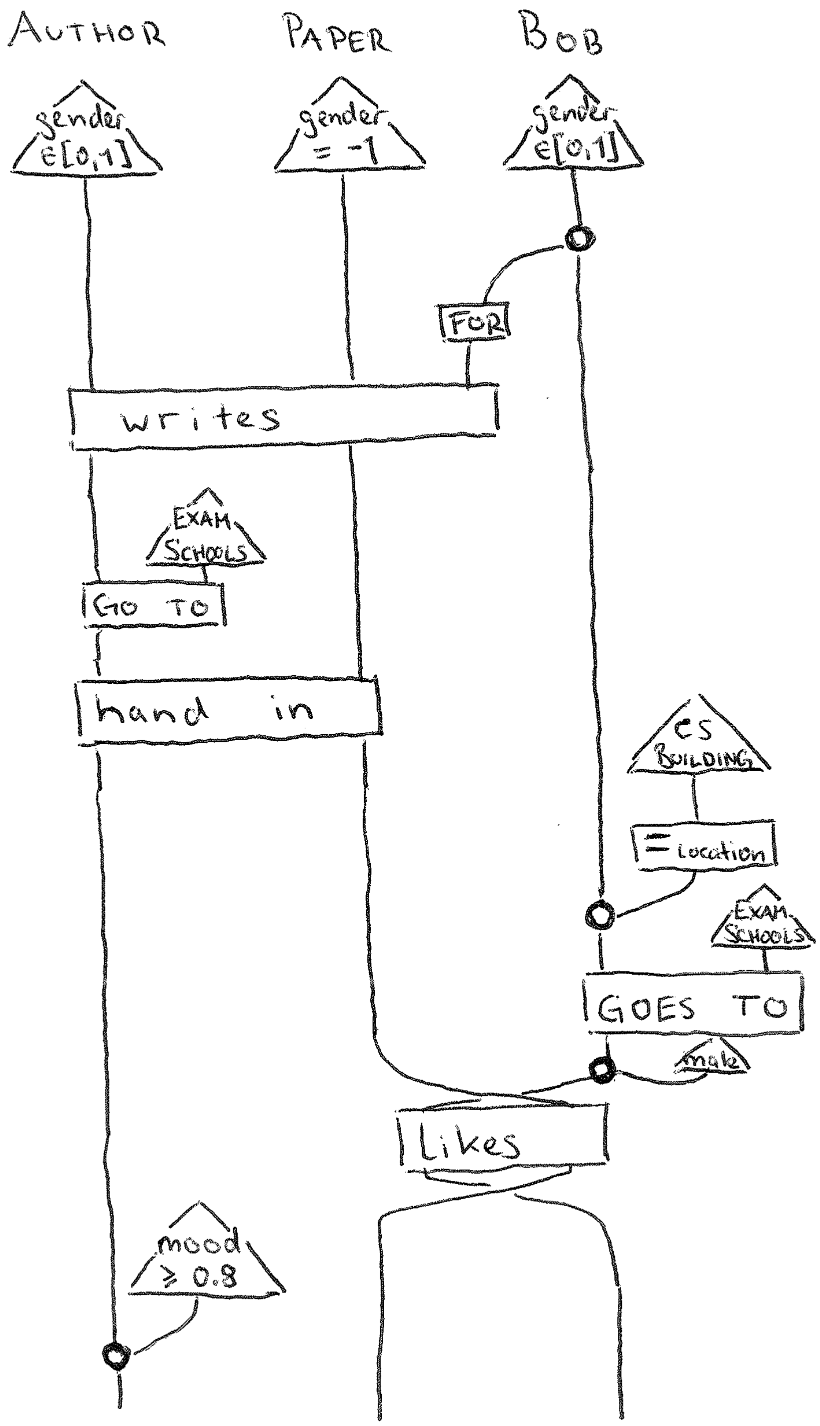
One such impossible case occurs when we assign “they” to Bob, as this sets his location to that of the Exam Schools. The third sentence, which requires him to be at the CS institute, therefore causes a contradiction for that case; the only diagram representing a nonempty state is now the one that identifies “they” with the author. The rest of the story is entirely unambiguous, thanks to Bob being the only subject in the third sentence that can be male (unlike the two buildings). A complete diagram, with everything fully resolved, can therefore be found on the next page. Bob thinks this is a great ending.

6 Conclusion

We have seen that a framework based on conceptual spaces is very suitable for text analysis. While some of the ideas from this paper can also be applied to the vector-based DisCoCirc model, pronoun resolution in particular relies on the intersective nature of gender adjectives, something that is hard to replicate in a distributional framework.

Over the last decade, several previous works have provided internal wirings for certain classes of words. However, there is clearly a limit to what can possibly be expressed within a purely compositional model of meaning, and it seems we have reached that limit now. Further open questions, such as the ones asked in Section 3.5, may have definite answers that lie outside the possibilities of any such framework.

As meaning evolves during a text, so does our way of understanding it. Taking context and associations into account, the same concept might represent two entirely different nouns. Therefore, a rigid model might never be able to accurately encompass every aspect of the English language. But while the cognition-based approach presented in this short paper is only an approximation, it brings us a lot closer to our goal.


Author
Paper
Bob
gender ∈[0,1]
gender = -1
gender ∈[0,1]
FOR
writes
EXAM SCHOOLS
Go to
hand in
CS BUILDING
=location
EXAM SCHOOLS
GOES TO
male
likes
mood ≥ 0.8

References

[1] Joe Bolt, Bob Coecke, Fabrizio Genovese, Martha Lewis, Dan Marsden, and Robin Piedeleu. Interacting Conceptual Spaces I: Grammatical Composition of Concepts. *arXiv:1703.08314*, 2017.

[2] Stephen Clark, Bob Coecke, and Mehrnoosh Sadrzadeh. Mathematical Foundations for a Compositional Distributional Model of Meaning. *arXiv:1003:4394*, 2010.

[3] Stephen Clark, Bob Coecke, and Mehrnoosh Sadrzadeh. The Frobenius Anatomy of Word Meanings I: Subject and Object Relative Pronouns. *arXiv:1404.5278*, 2014.

[4] Bob Coecke. The Mathematics of Text Structure. 2019.

[5] Bob Coecke, Martha Lewis, and Dan Marsden. Internal Wiring of Cartesian Verbs and Propositions. *arXiv:1811.05770*, 2018.

[6] Edward Grefenstette and Mehrnoosh Sadrzadeh. Concrete Models and Empirical Evaluations for the Categorical Compositional Distributional Model of Meaning. *Computational Linguistics*, 41:71–118, 2015.

[7] Peter Gärdenfors. *Conceptual Spaces: The Geometry of Thought.* MIT Press, 2000.

[8] Nicola Pinzani. A Metaphysical Composition. 2018.

[9] Brian Tyrrell. Applying Distributional Compositional Categorical Models of Meaning to Language Translation. *arXiv:1811.03274*, 2018.