A Tale of Deals, No Deals and Baked Beans

Irene Rizzo

Hilary 2019

Contents

1 Introduction

In this project, we propose a DiscoCat and DiscoCirc Model of Meaning based on a modified conceptual space. Conceptual Spaces were firstly introduced by Gärdenfors in [10] and have been extensively used by B. Coecke et al. in [1] as a way to model language starting from cognitive assumptions rather than linguistic ones. DisCoCirc has been recently introduced by Coecke in [6], and consists of an extension of the Distributional Compositional Categorical Model of Meaning (DisCoCat) constructed by Coecke et al. in 2010 [7]. DisCoCirc, a part from having many (disco)cats, combines sentences in a circuit, in order to model text. Moreover, in DiscoCirc sentences are processes, which upgrade the knowledge about the subjects and objects.

The meaning space we propose is based on **Rel**. However, the noun space will be constructed generalising the conceptual space described in [1]. Specifically, we will make use of bipruducts to link different semantic areas. In general, we will lose the property of convexity, but we will see that some types of nouns will still correspond to convex subsets. The gained flexibility will allows to define more complicated words, that would be challenging to define purely conceptually. Furthermore, we will define a notion of graded entailment and of negation.

Overall, the construction of the noun space will consists of "stacking up" different semantic area spaces, which are essentially the concept spaces described in [1]. The idea is to create semantic classes, inside which nouns are modelled conceptually, and then use these classes as a sort of *dictionaries* from which we can construct other nouns. In other words, we will be able to construct new nouns from older nouns.

This approach reflects a more profound belief regarding language and cognition. If we observe the cognitive process of learning how to speak a language, we notice that we usually start with incorporating very basic words corresponding to very practical concepts. For example "mum", "food", "poo" , or, in case of English being a second language, we probably are going to learn "beer" first. Then, the learning of new words often relies on pre-knowledge of simpler words. For example, let's suppose that Alice is learning English and that she doesn't know the word "Ale" yet; then Alice goes to the pub and the waiter asks "What Ale would you like, Madame?", to which she will probably reply "Excuse me, what does it mean?". Then, the waiter will say "It is a kind of beer.. such and such..". At this point, Alice will have learned a new word, the meaning of which was explained using the word "beer", which she already mastered.

Moreover, if we imagine the evolution of language itself, it is reasonable to assume that language initially started as a collection of words indicating very practical concepts, like the Buffalo we are trying to hunt, or the storm that is coming. Then, human lifestyle kept evolving and humans became more and more sophisticated: as their immediate primal needs became satisfied by increasing technology, and some spare free time become available, they started to think and care about "secondary" stuff. The more complex the lifestyle and the human perception of reality, the more complex the concepts needed to describe them. Thus, language stacked up more and more words, created combining existing concepts, broadening, narrowing, abstracting them and even inventing imaginary ones like "unicorns".... We have made it to the present time, in which language has allowed for the creation of the absolute protagonists of the Internet: memes!

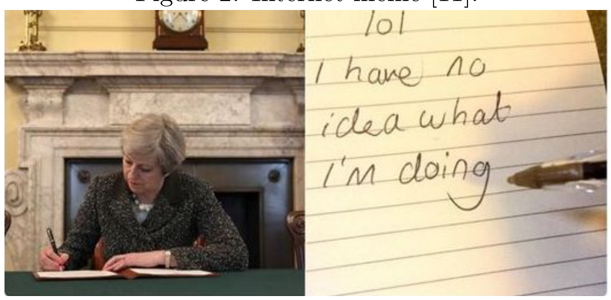
After modelling the necessary ingredients, we will tell a short story about Dave, Terry and Brexit, using DisCoCirc on our constructed noun space. Each sentence will be firstly modellled separately, using a modified DisCoCat, and the vocabulary will almost entirely be derived from "memes" from the Internet.

A disclaimer: the author takes no responsibility for any potential offence regarding political beliefs! The author is simply letting the Internet God of Memes speak and have a laugh! As a further disclaimer: the majority of semantic examples produced throughout the project are not meant to be rigorous, whilst they definitely are the proof of food being the constant thought in the author's mind.

Now, let us begin.

Figure 1: Internet meme [11].

Figure 2: Internet meme [11].



2 Modified DisCoCat

As mentioned in the introduction, the model in which we will tell our tale, will be DisCoCirc [6]. However, we will also provide the internal structure of each sentence using DisCoCat [7], where the traditional sentence wire is replaced by noun wires, so that it is compatible with DisCoCirc. More importantly, it is worth to point out that in this project, **all** wires will be of type N, which indicates the **noun space**.

The Grammar Category will be the usual pregroup, detailed by Coecke et al. in multiple publications (e.g. [1]) and firstly developed by Lambek in [9]. We often won't give full details of the underlying grammatical types of the modelled words, for our focus will mainly be on the meaning category. More details on the grammar can be found in the the literature.

In this section we give a brief overview of the representation of different types of words as processes in the meaning category.

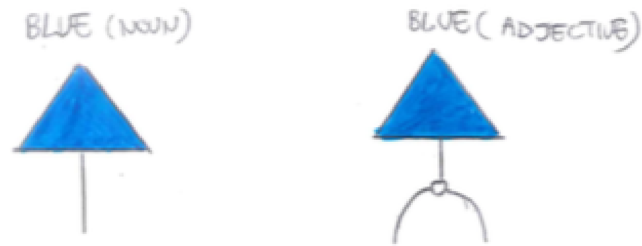
Aside on notation: throughout the project, we will represent the meanings of words in black ink and the from-meaning-of-words-to-meaning-of-sentence map in blue ink.

2.1 Nouns and Adjectives

Nouns have grammatical type n, thus, in the meaning category they corresponds to states of the system N.

Adjectives have grammatical type nn^l. Thus, in the meaning category they are states of $N \otimes N$. In this project we will only consider **intersecting adjectives**, which are detailed by Coecke et al. in [1, Section 5.1.2]. These are obtained by applying a noun to a spider. An example is given below.

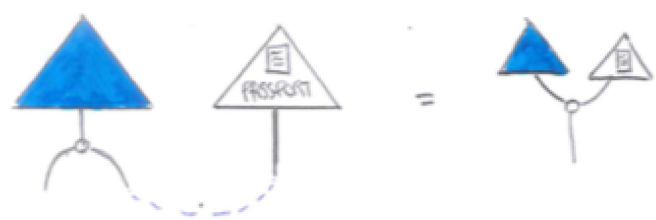
Figure 3: Intersecting adjective "Blue".



adjective.png

This type of adjectives is "intersecting" because, when applied to a noun, the resulting meaning is given by the intersection of the meanings of two nouns:

Figure 4: A blue passport

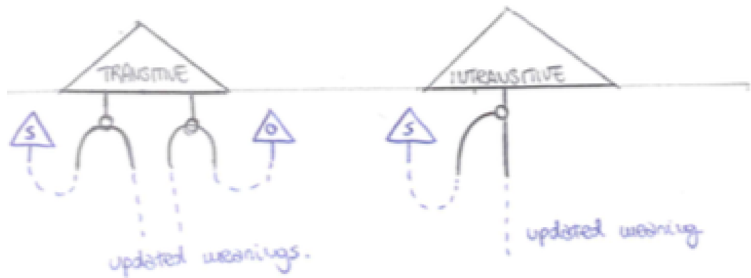


passport.png

2.2 Verbs

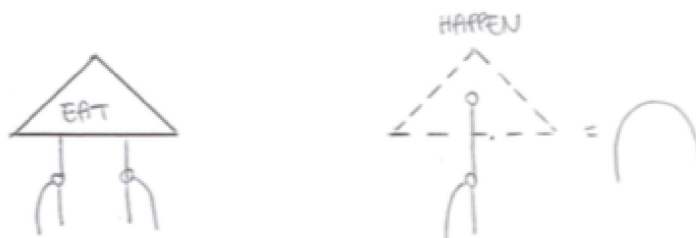
In the traditional DiscoCat ([7]) transitive verbs are given grammatical type $n^r s n^l$, where s is the sentence type. Here, however, we adopt the construction given by Coecke in [6]. In particular, in this project we will only consider the modified version of semi-cartesian verbs described in [6]. In the latter, the sentence type is constructed from noun types. Such construction reflects the idea that verbs are maps that update the meaning of nouns. More specifically, transitive sentences (i.e. the ones obtained from transitive verbs) have type $s = nn$. As a consequence, transitive verbs have type $n^r nnn^l$ and thus, they are states of $N \otimes N \otimes N \otimes N$ in the meaning space. Similarly, intransitive sentences have type $s = n$ and intransitive verbs are of type $n^r n$, i.e. they are states of $N \otimes N$. Under this interpretation, transitive verbs update the meaning of the subject noun and of the object noun, while intransitive verbs update the meaning of the subject nouns only.

Figure 5: Transitive vs Intransitive verbs



Some examples are given below.

Figure 6: Examples of transitive and intransitive verbs.

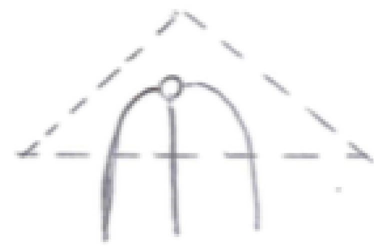


examples.png

Note that the internal wiring of the intransitive verb `to happen`, is such that the state is equivalent to a cap. Indeed, this verb does not really add any extra information about the meaning of the noun it is applied to. In fact, the only information that it encodes is that something has come to realisation. Thus, it is reasonable to model is as cap.

One particular case of intransitive verb is the verb `to be`. We model it as done by Coecke in [6] and thus, we assign it grammatical type $n^r n n^l$. Therefore, `to be` is a state of $N \otimes N \otimes N$. Note that, like `to happen`, the verb `to be` has a wire-only structure, as we can see below.

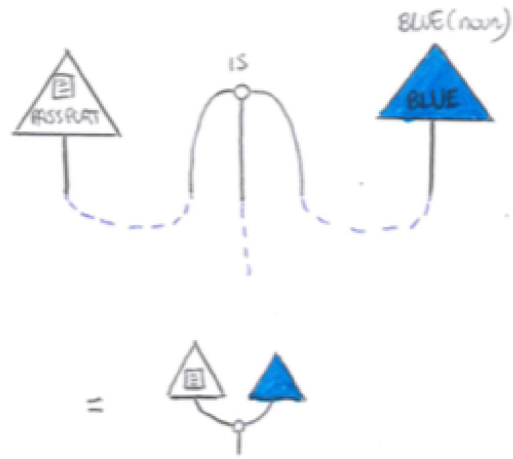
Figure 7: A special case: "To Be"



be.png

As we can observe below, this construction allows us to model the sentence `the passport is blue` in the modified DisCoCat, such that the resulting meaning is the same as the one obtained from `a blue passport`.

Figure 8: "The passport is blue"



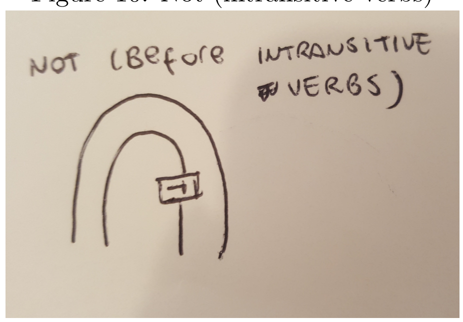
is blue.png

2.3 Logical function words and Relative Pronouns

In the next sections we will often encounter negations and negative sentences. As we will see, negations arise from the presence of words such as `not` or `no`, placed before verbs or nouns/intersecting adjectives. We model such logical function words as follows, depending whether it is placed before a transitive verb, intransitive verb or a noun/intersecting adjective.

Figure 9: Not (transitive verbs) and No (nouns/adjectives)

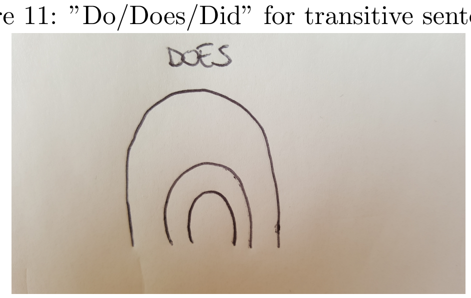
Figure 10: Not (intransitive verbs)



Note that the state in Figure 10 has the same wiring as the "not" word modelled in [7]. Moreover, note that the $\neg$ box, which we define in Section 6, is not a process in the theory and it has to be thought of as an extra added feature.

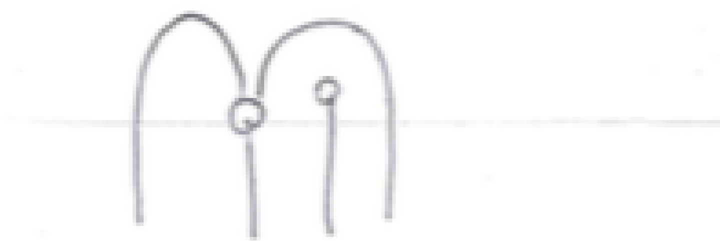
An other logical function word we encounter in interrogative or negative sentences (e.g. "Dave did not say Brexit"), is `do/did/does`. If `do` appears in an intransitive sentence, we model it as a double nested cap, adopting the construction given in the literature, e.g. in [7]. In transitive sentences, we model it a triple nested cap, in order to account for the sentence type being $s = nn$. An example of the latter is given below.

Figure 11: "Do/Does/Did" for transitive sentences.



Finally, in Figure 12 we recall the representation of the subject relative pronoun given by Clark et al. in [8].

Figure 12: Subject Relative Pronoun in intransitive sentence

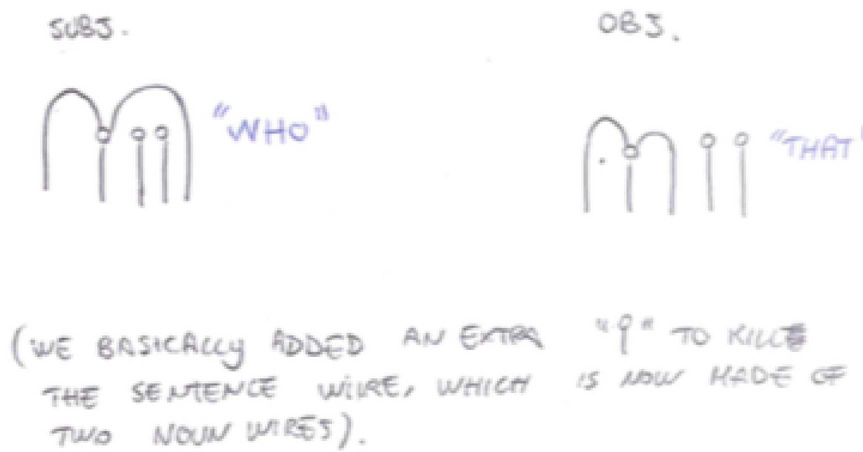


pronoun intransitive.png

By considering all the wires to have noun type, we can use this representation to indicate those subject relative pronouns which are followed by an intransitive verb, e.g. "The Dog **that** barks".

On the other hand, we define the subject and object relative pronouns for transitive sentences, by slightly modifying the construction given in [8], as follows.

Figure 13: Modelling relative pronouns.



pronouns.png

We have now given a brief overview of how we can model words in the modified DiscoCat. We now move on to construct the noun space N. In order to do so, we first recall the notion of conceptual space.

3 Background: Conceptual Spaces

In this section we recall the notion of conceptual spaces and their underlying algebraic structure.

As detailed in [1], conceptual spaces are structures based on quality dimension, such as colour, shape, taste etc... Some dimensions are *separable*, i.e. they do not interact with each other, and some others are *integral*, i.e. the assignment of a value in one dimension is linked to the assignment of a value in an other dimension. A set of integral dimensions is called **domain**. Then, a conceptual space arises from different domains interacting with each other via some categorical structure.

Conceptual spaces are usually constructed so that they admit the property of *convexity*. Indeed, as explained by Coecke et al. in [1] and by Marsden et al. in [3], given a meaning space for some concepts, intuitively we would assume that if two different points represent the same concept, then all the points in between do as well (i.e concepts should be closed under taking mixtures). This property corresponds exactly to convexity. In mathematical terms, the algebraic structure of convexity derives from the monad of finite probability distributions. Let us recall the notion of a monad.

Definition 3.1. A monad on a category $\mathcal{C}$ is a triple (T, η, μ), where T is an endofunctor, and $\eta : 1_{\mathcal{C}} \Rightarrow T$ and $\mu : T \circ T \Rightarrow T$ are natural transformations representing respectively the monad unit and the monad multiplication. These satisfy the following associativity and unitality conditions:

- $\mu \circ T\mu = \mu \circ \mu T$
- $\mu \circ T\eta = \mu \circ \eta T = Id_T$

Monads induce algebraic structures as follows:

Definition 3.2. Given a monad T on a category $\mathcal{C}$, a T-algebra is a pair (A, α), where A is an object of $\mathcal{C}$ and $\alpha : TA \to A$ is a $\mathcal{C}$-morphism satisfying:

- $\alpha \circ \eta_A = Id_A$
- $\alpha \circ T\alpha = \alpha \circ \mu_A$

These algebras form a category, namely the *Eilenberg-Moore category* $\mathcal{C}^T$, whose objects are the said algebras and morphisms are those $\mathcal{C}$-morphisms which commute with the algebraic operation.

Now let us go back to convexity. Let X be a set, then we can form convex combinations of elements of it, as follows.

Definition 3.3. A convex combination of elements of X, is a sum $\sum_i p_i |x_i\rangle$, such that each p_i is a positive real and $\sum_i p_i = 1$.

Hence, a convex combination is a finite probability distribution on X. The monad of finite probability distribution is given by the endofunctor D mapping X to its set of convex combinations, DX. Then, a *convex algebra* (A, α) is a D-algebra, which we will consider acting in **Rel**. Then, **ConvexRel** is the Eilenberg-Moore category of these convex algebras. The objects of $ConvexRel$ are the convex algebras and the morphisms are the **convex relations**. The latter are the convex algebraic closure of the usual relations.

Definition 3.4. A relation $R : A \to B$ is a convex relation if the following condition holds.

$$R(a_1, b_1) \wedge R(a_2, b_2) ... \wedge R(a_n, b_n) \Rightarrow R\bigg(\alpha(\sum_i p_i |a_i\rangle), \alpha(\sum_i p_i |b_i\rangle)\bigg)$$

In more λ-calculus terms, this means that convex relations satisfy the extra "mixture" rule given by the following deduction:

$$\frac{\forall i. a_i \mapsto b_i}{\alpha(\sum_i p_i |a_i\rangle) \mapsto \alpha(\sum_i p_i |b_i\rangle)} \; a_i : A, b_i : B$$

As stated in [1, Theorem 1], **ConvexRel** is a compact closed category and it is used by Coecke et al. in [1] to model conceptual spaces. The latter is constructed as an object in **ConvexRel** corresponding to the tensor product of different quality domains. The nouns are then convex subsets of the conceptual space.

In the next section we will extend this model and construct our noun space N.

4 The Meaning Space

In this section we draw from the above theory and we construct our model for the noun space N. Our construction will lose some of the convexity properties, but will allows us to model a notion of orthogonal negation and incorporate more semantically different nouns in the same space.

4.1 Area spaces

Following from [1], we first note that we can classify words into **semantic areas**. Words in each area can usually be described by the same kind of qualities. For example, in order to talk about *food*, we need to give values for taste, colour, texture, smell; on the other hand, these qualities are not all suitable to talk about *animals*. Therefore, we define a semantic **area space** A to be an object in **ConvexRel**, such that it is the tensor product of those domains D_i that are necessary and sufficient to describe words in the corresponding semantic area. For example

$$A_{food} = D_{taste} \otimes D_{colour} \otimes D_{texture}$$

$$A_{animal} = D_{colour} \otimes D_{cuteness} \otimes D_{size}$$

We define **atomic nouns** to be states of a system A_i, i.e. convex subsets. It is worth pointing out that "atomic" refers to an "atomic meaning" that refers to a single semantic area. The definition of atomic nouns corresponds to the definition of nouns given in [1].

Example 4.1. *bakedbeans* $= I \to A_{food}$

One limitation is that only very few nouns can be defined directly using qualities, while for many others it is easier to define them using other nouns. Therefore we decide to sacrifice the convexity property for more general nouns and extend the space as follows below.

4.2 Noun space

Let us temporarily forget about convexity, with the promise that some of it will be recovered later. Let us construct a copy of these area spaces A_i as objects of **Rel**, so that we have more relations allowed. Let the **noun space** N be an object of **Rel**, such that:

$$N = \bigoplus_i (A_i)$$

Then, we define **nouns** to be the states of the system N in **Rel**, i.e. subsets (not necessarily convex subsets) of $I \times N \cong N$. Note that we can recover the original atomic nouns in this new space. Indeed, atomic nouns in N are given by the inclusions of the original atomic nouns into N. For example, let

$$N = A_{food} \oplus A_{drinks} \oplus A_{weather} \oplus A_{animals} \oplus A_{travel-documents}$$

and *bakedbeans* $= I \to A_{food}$ as given in the previous example. Them, we can define `bakedbeans` $: I \to N$ to be

$$\begin{aligned}\texttt{bakedbeans} =& (i_1 \circ F(bakedbeans)) + (i_2 \circ 0_{I,A_2}) + (i_3 \circ 0_{I,A_3}) + (i_4 \circ 0_{I,A_4}) + (i_5 \circ 0_{I,A_5}).\\ =& (i_1 \circ F(bakedbeans)) + 0_{I,N} + 0_{I,N} + 0_{I,N} + 0_{I,N}\\ =& (i_1 \circ F(bakedbeans)).\end{aligned}$$

where the indices refer respectively to *food, drinks, weather, animals, travel-documents*, the $i_j : A_j \hookrightarrow N$ are the injective maps of the biproduct, the 0 indicate the zero maps (the empty relation) and F : **ConvexRel** $\to$ **Rel** is the forgetful functor. Thus, in this model `baked beans` is the image in N of a convex subset of A_{food}. This construction allows us to recover some aspects of convexity when dealing with atomic nouns. The model is essentially operating on two levels:

1. the **area space**, which is smaller, conceptual and has convexity and can be used when we are dealing with nouns from the same semantic area;
2. the **noun space**, which expands over different semantic areas, it does not preserve convexity but allows for some extra flexibility, as we will see below.

Having recovered atomic nouns, we can now recursively build new nouns as superpositions (in **Rel** these are unions of subsets) of given nouns. Let us observe this in following example, which is highly motivated by Fig.14.

Figure 14: The food area [11].

Example 4.2. Let

$$N = A_{food} \oplus A_{drinks} \oplus A_{weather} \oplus A_{animals} \oplus A_{travel-documents}.$$

as before. Then, let us consider the nouns `baked beans, tea, ale, rain, corgi, blue passport`. Then, the following nouns are constructed as follows

$$\begin{aligned} \texttt{Britain} :=& (\texttt{bakedbeans} + \texttt{tea} + \texttt{ale} + \texttt{rain} + \texttt{corgi}) \\ \texttt{Brexit} :=& (\texttt{Britain} + \texttt{bluepassport}). \end{aligned}$$

4.2.1 Adjectives

Following from the construction of nouns, we can construct intersecting adjectives as follows. Let us recall the notion of a property $P_{property}$, which was defined in [1, Section 5] as a convex subset of a quality domain. For example, P_{blue} is the convex subset of the colour domain D_{colour} corresponding to "blue".

Let us first consider the convex subset corresponding to the whole A_{food}, and let $food$ be the state of the system A_{food} corresponding to that subset. Then, the noun `food` is given by the state $i_{food} \circ F(food)$. In this way, we can think of the area space A_{food} to essentially be the noun `food`. Let us now construct the noun `blue` through an example.

Example 4.3. For the sake of simplicity, let us assume that the only area spaces, in which the colour domain appears, are `food`, `animals` and `travel-documents`. Then the noun `blue` is given by

$$\texttt{food}/P_{blue} + \texttt{animals}/P_{blue} + \texttt{travel} - \texttt{documents}/P_{blue}.$$

where $\texttt{noun}/P_{blue}$ indicates the area space, the D_{colour} component of which has been restricted to the colour blue. Then let $\texttt{food}/P_{blue} = \texttt{blueberry}$, $\texttt{animals}/P_{blue} = \texttt{blue} - \texttt{jay}$ and $\texttt{travel} - \texttt{documents}/P_{blue} = \texttt{blue} - \texttt{passport}$. Then $\texttt{blue} = \texttt{blueberry} + \texttt{bluejay} + \texttt{blue} - \texttt{passport}$.

This example shows how this construction mirrors the idea that the "concept of blue" abstracts and includes all the things that are blue.

The intersecting adjective corresponding to the noun $\texttt{blue}$, will simply be $copy_N \circ \texttt{blue}$ as seen in Figure 3. Note that *copy* is an instance of the *spider* defined for **Rel** in Example 7 of [1].

5 A notion of entailment

This model admits an immediate notion of entailment, which corresponds to the non-graded entailment definition given by D. Bankova et al. in [4, Section 4]. We have that $\texttt{noun}_1$ *entails* $\texttt{noun}_2$ if and only if $\texttt{noun}_1 \subseteq \texttt{noun}_2$. For example, if we define `food` as above and we consider the nouns `baked beans, cinnamon rolls, pizza, croissant` as being subsets of A_{food}, then we can easily see that `baked beans` entails `food`. But we can do better than this! In fact, we can add some notion of graded entailment, in a similar way as it was developed in similarly [4].

5.1 Partial and graded entailment

Let us go back to the noun space example given above, i.e $N = A_{food} \oplus A_{drinks} \oplus A_{weather} \oplus A_{animals} \oplus A_{travel-documents}$. Consider the biproduct projection maps $\pi_i : N \to A_i$, with $i \in I$ and

$$I = \{food, drinks, weather, animals, travel - documents\}.$$

Then we can define *partial entailment* as follows:

Definition 5.1. We say that $\texttt{noun}_1$ *partially entails* $\texttt{noun}_2$, if and only if there is a least an $i \in I$ such that

$$\pi_i \circ \texttt{noun}_1 \subseteq \pi_i \circ \texttt{noun}_2$$

Let us write $\preceq$ for the partial entailment and $\subseteq$ for the "total" entailment. Clearly if $\texttt{noun}_1 \subseteq \texttt{noun}_2$ AND $\texttt{noun}_2 \subseteq \texttt{noun}_1$ then the two nouns have the same meaning in the semantic model. However, we can have $\texttt{noun}_1 \preceq \texttt{noun}_2$ AND $\texttt{noun}_2 \preceq \texttt{noun}_1$, without necessarily the two nouns being synonyms in the model.

The notion of partial entailment then leads us to define a "grade" of entailment. Let n be the number of area spaces in our noun space and let $\texttt{n}_1$ and $\texttt{n}_2$ be two nouns. Then, let us consider the functions $f_i : A_i \times A_i \to \mathbf{2}$, where $\mathbf{2} := \{0, 1\}$, given by

$$f_i = \begin{cases} 1 & \text{if } \pi_i \circ \texttt{n}_1 \subseteq \pi_i \circ \texttt{n}_2 \\ 0 & \text{if } \pi_i \circ \texttt{n}_1 \nsubseteq \pi_i \circ \texttt{n}_2 \end{cases}.$$

Then, we say that $\texttt{n}_1$ partially entails $\texttt{n}_2$ with grade k, written $\texttt{n}_1 \preceq_k \texttt{n}_2$, where k is defined as

$$k = \frac{1}{n} \sum_{i=1}^{n} f_i.$$

If $k = 0$ clearly there is no entailment, and if $k = 1$ then $\texttt{n}_1 \subseteq \texttt{n}_2$.

6 Negation

It is very desirable to also add a notion of negation. Indeed, negation often appears in sentences, usually as negation of the predicate, as in "Dammit, I said I wanted breakfast, I *did not say* I wanted Brexit!" (Figure 1). In this case, the negation of the verb implies that the action did not happen, so it refers to a *lack* of action. This lack of action corresponds to one of many valid ideas of sentence negation. Indeed, we could also think of the negation of "I said I wanted Brexit" a being "(someone else, not me) said they wanted Brexit" or "I said something else". In these cases, it is the subject or object to be negated rather than the predicate. These notions are all valid logical negations of the whole sentence, since if we assign $\top$ to $\neg$"I said I wanted Brexit", then we are just asserting that the exact action of "me-having said-that specific thing" is false (assuming classical logic). However the falsity could come from any of the components.

Logic discussions aside, since all our wires are noun types, it is reasonable to define noun negation, as in "Deal or *No Deal*?" (Figure 15). In this section, we focus on negating **meanings**, and we adopt the convention that sentence negation coincides with negating the predicate...at the end of the day, "action counts more than words!"

We define meaning negation by adapting D. Widdows' orthogonal negation described in [5], which relies on quantum-logic ideas and it's motivated by document retrieval tasks. We share Widdows' intuition that the negation of a concept should be orthogonal to the negated concept. In *Rel*, subsets are orthogonal if their intersection is empty, therefore we will construct negations, such that this property is respected. In the case of states being subsets, Widdows defines negation as the complement of the subset. However, we want to avoid the situation in which, by taking the complement of the empty set in an area space, we get the whole area space. Let us proceed to defining **meaning negation** for nouns.

Figure 15: A noun v. its negation [11].

6.1 Negating nouns

Our approach consists of negating concepts component-wise in the relevant area spaces. Let $(-)^\circ$ be an operator on the states that returns the "complement state". We know that a state of a system A is a subset $R \subseteq I \times A$, such that $(*, a) \in R$ if and only if $* \sim a$. Then we define $(R)^\circ$ such that $(*, a) \in (R)^\circ$ if and only if $(*, a)$ is not in R. Let us explain negation for an atomic nouns through an example.

Example 6.1. Let `baked beans` be an atomic noun in A_{food}, then its negation, written $\neg$`bakedbeans`, is given by

$$\neg\texttt{bakedbeans} = i_{food} \circ (\pi_{food} \circ \texttt{bakedbeans})^\circ$$

Note that $\neg$`bakedbeans` is state of N (thus it is a legitimate noun), and that both $\neg$`bakedbeans` and `bakedbeans` live in the same semantic area. Furthermore, by considering them as subsets of A_{food}, we can easily verify that $\neg\texttt{bakedbeans} \cap \texttt{bakedbeans} = \emptyset$ and $\neg\texttt{bakedbeans} \cup \texttt{bakedbeans} = A_{food}$; hence, the meanings are orthogonal and complementary.

Unfortunately, $\neg$`bakedbeans` is not an atomic noun (negation does not preserve convexity), however it has a very desirable property: it indicates "all the **food** that is not baked beans" rather than some other concepts that, despite of being orthogonal to `baked beans`, refers to non-foody words like `pink elephants`. This issue would arise if we were simply taking the complement of the states as subsets of N. This construction mirrors the intuition that often, when we negate a concept, we don't want to exit the semantic area, since we are probably still talking of related stuff. In fact, if I order "a full English breakfast, no baked beans please", it doesn't mean that I don't want breakfast at all, or that I want my breakfast to include pink elephants. It simply means that I want all the items of the full breakfast except for the beans, and without weird additions.

We can generalise the above construction to all nouns, as follows. We want to avoid empty relations to be mapped to the "full" relation by the complement; hence, we have to make the complement operator more specific. Let

$$(R)^{\perp} = \begin{cases} (R)^{\circ} & \text{if } R \neq 0 \\ R & \text{if } R = 0 \end{cases}.$$

where 0 is the empty relation. Now we can define the following.

Definition 6.1. Let $\mathtt{n}$ be a noun in a noun space $N = \bigoplus_j A_j$ with $j = 1, ..., n$, then we define the operator $\neg$ as follows

$$\neg\mathtt{n} = \sum_{j=1}^{n} i_j \circ (\pi_j \circ \mathtt{n})^{\perp}.$$

It can be easily seen that the definition given for atomic nouns is a special case of the above. By this definition, negations are states of the system N, which live in the same semantic areas of the negated noun.

Example 6.2. Following from the previous example, let, for the sake of simplicity, A_{food} be spanned by disjoint nouns `cinnamon rolls`, `pizza`, `baked beans`, `croissant`. Taking inspiration from Figure 14, let us consider the noun `Deal` to live in the food space, i.e. let us momentarily forget about cheap holidays to Spain and all the other things we wish a good Deal includes. If

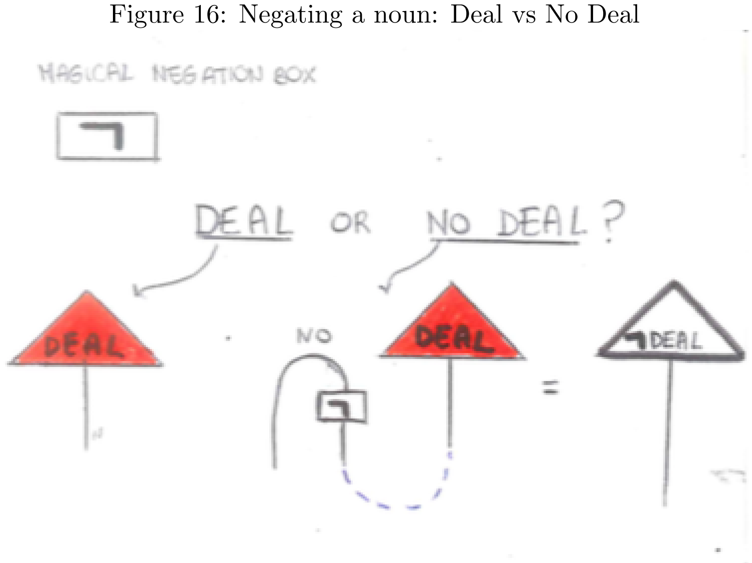
$$\mathtt{Deal} = \mathtt{cinnamonrolls} + \mathtt{croissant} + \mathtt{pizza}.$$

then we obtain

$$\neg\mathtt{Deal} = \mathtt{bakedbeans}.$$

Before proceeding to using negation more freely, it is a good idea to define the operator $\neg$ as a box in the graphical calculus. We note that this box is NOT a process in the process theory, but more like a magical box that operates on states by mapping them to their negation. Since it is magical, we will also allow it to slide along the wires freely. Therefore, the diagrams in the modified DiscoCat are to be thought of as string diagrams + magical box. The negation box appears inside words like `no` and `not`, as we saw in the previous section. The action of the box is made explicit via the following example:

Figure 16: Negating a noun: Deal vs No Deal



Hence, the meaning of the string `no Deal` corresponds to the meaning of ¬`Deal`.

6.2 Negating adjectives

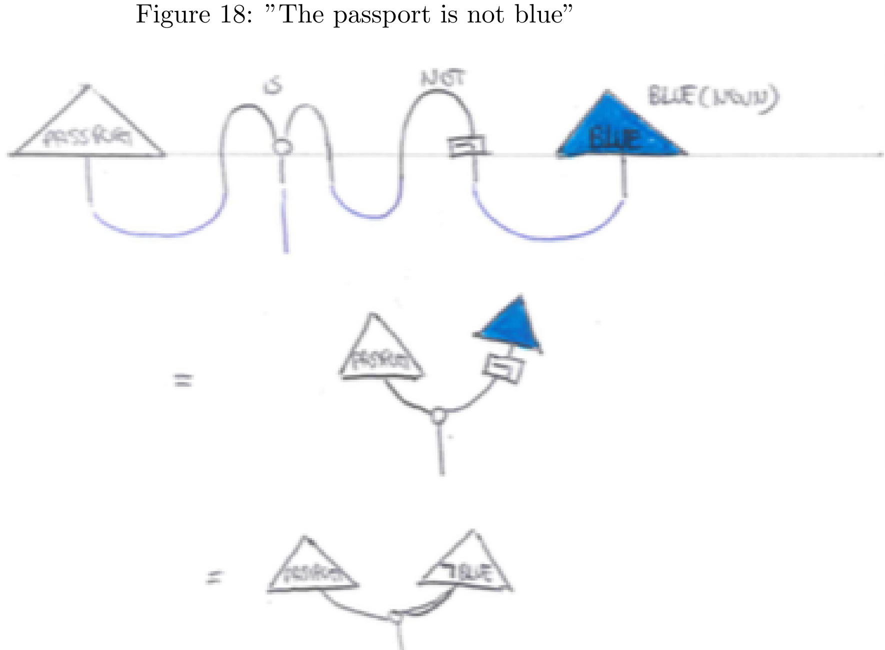
Now that the magic is unveiled, we can negate intersecting adjectives by simply negating the corresponding noun. For example:

Figure 17: Negation of intersecting adjective "Blue".

blue.png

From this definition, we expect that the sentence `the passport is not blue` corresponds to applying the intersecting adjective ¬`blue` to the noun `passport`. Indeed, in the modified DisCoCat we have the following.

Figure 18: "The passport is not blue"

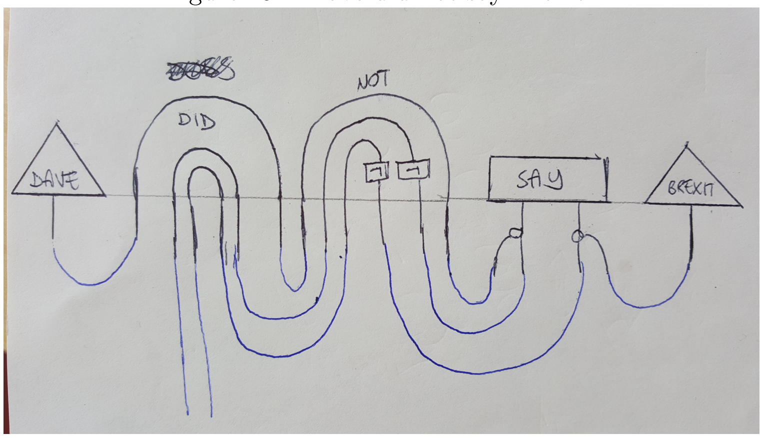


is not blue.png

6.3 Negating verbs and sentences

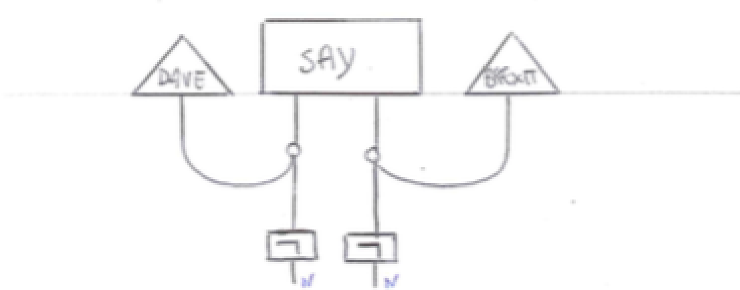
As seen in Figure 18, sometimes the meaning of a negative sentence[1] can be encoded by the meaning of a negative adjective. However, this is not always possible. Indeed, a sentence like `Dave did not say Brexit!` cannot simply be encoded by the negation of a noun or adjective. As seen in [7], we can negate the predicate, and consequently the whole sentence, by making use of the already mentioned "magical negation box". Let us make this explicit through an example, which features a transitive verb:

Figure 19: "Dave did not say Brexit"



This sentence clearly simplifies to the negation of the sentence `Dave said Brexit`, as we can see in Figure 20 below.

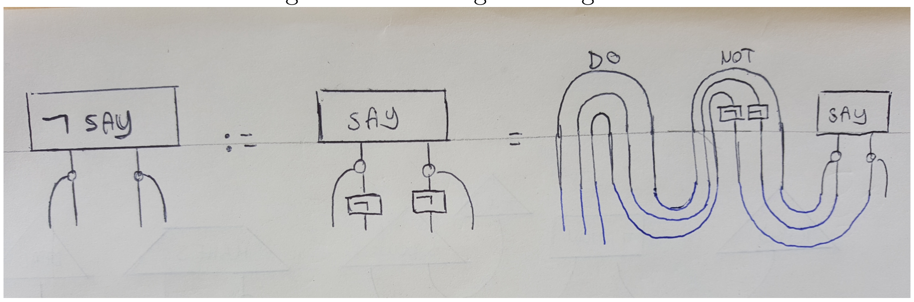
Figure 20: ¬("Dave said Brexit")



From the above, we can also notice that the negation of the sentence corresponds to the negations of the updated subject and object nouns. Then, we can define the negation of a verb to be the following:

[1]Note that, as mentioned earlier, we consider the negation of a sentence to be the negation of its predicate.

Figure 21: Defining verb negation



Note that when we combine sentences in the next sections, we will often use the form ¬`verb`, instead of `does` + `not` + `verb`. Now let us do some DisCoCats!

7 The Tale

In the previous sections we have prepared all the essential ingredients to tell our short tale, which is inspired by Internet memes about Brexit, such as Figure 1 and 2. The tale is the following.

`Dave said breakfast. Dave didn't say Brexit!...Britain heard Brexit. Terry, who doesn't know what she is doing, makes a Deal. Britain doesn't like the Deal that Terry made. No Deal, which isn't fun, happens...Oops!`

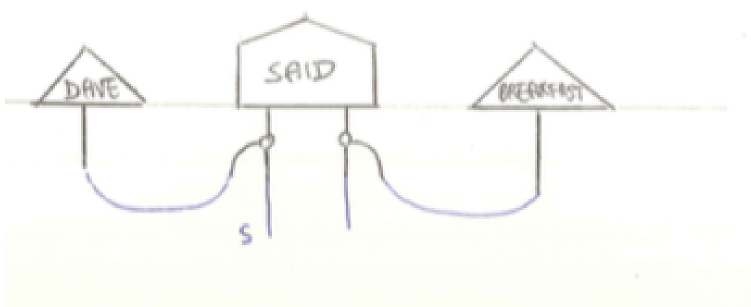
Here, we will first extract each sentence meaning separately, using the modified DisCoCat, then we combine them together using DisCoCirc. Note that. although these models are assumed to be based on the constructed meaning space, the words meanings won't be explicitly assigned values from the meaning space: we leave this to future development.

7.1 Single Sentences

In this section we extract the meaning of each sentence using the modified DiscoCat model. As previously mentioned, we will draw the states corresponding to word meanings in black ink, and the from-meaning-of-words-to-meaning of sentence map in blue ink. The output of each sentence will be either a noun (updated meaning of the subject) or two nouns (updated meaning of the subject and object).

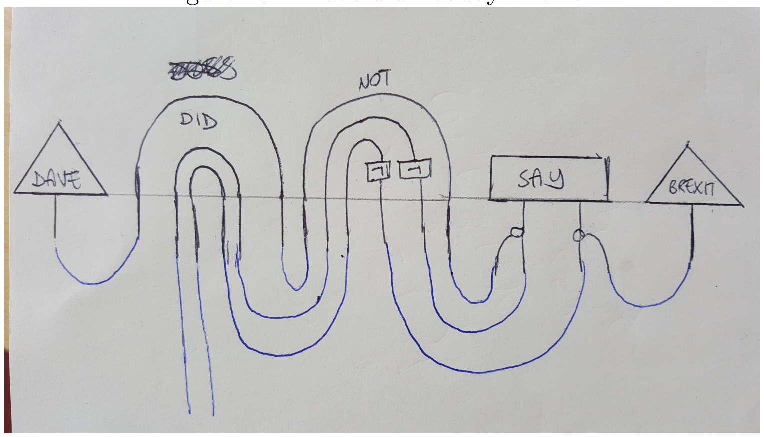
`Dave said breakfast` corresponds to Figure 22 below.

Figure 22: "Dave said breakfast"



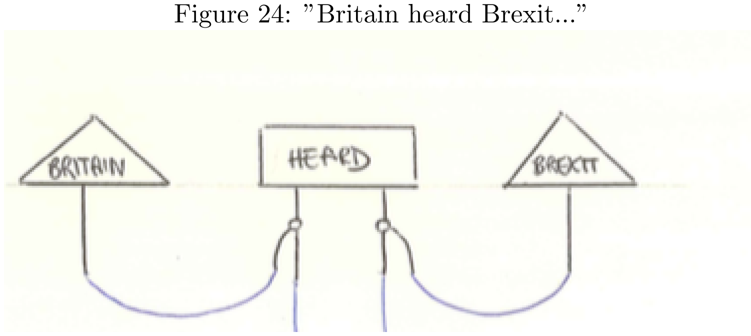
`Dave did not say Brexit!` is given by Figure 23 below. Note that `did not say` could be directly replaced by ¬`say`, as seen in Figure 21.

Figure 23: "Dave did not say Brexit!"



`Britain heard Brexit` corresponds to Figure 24 below.

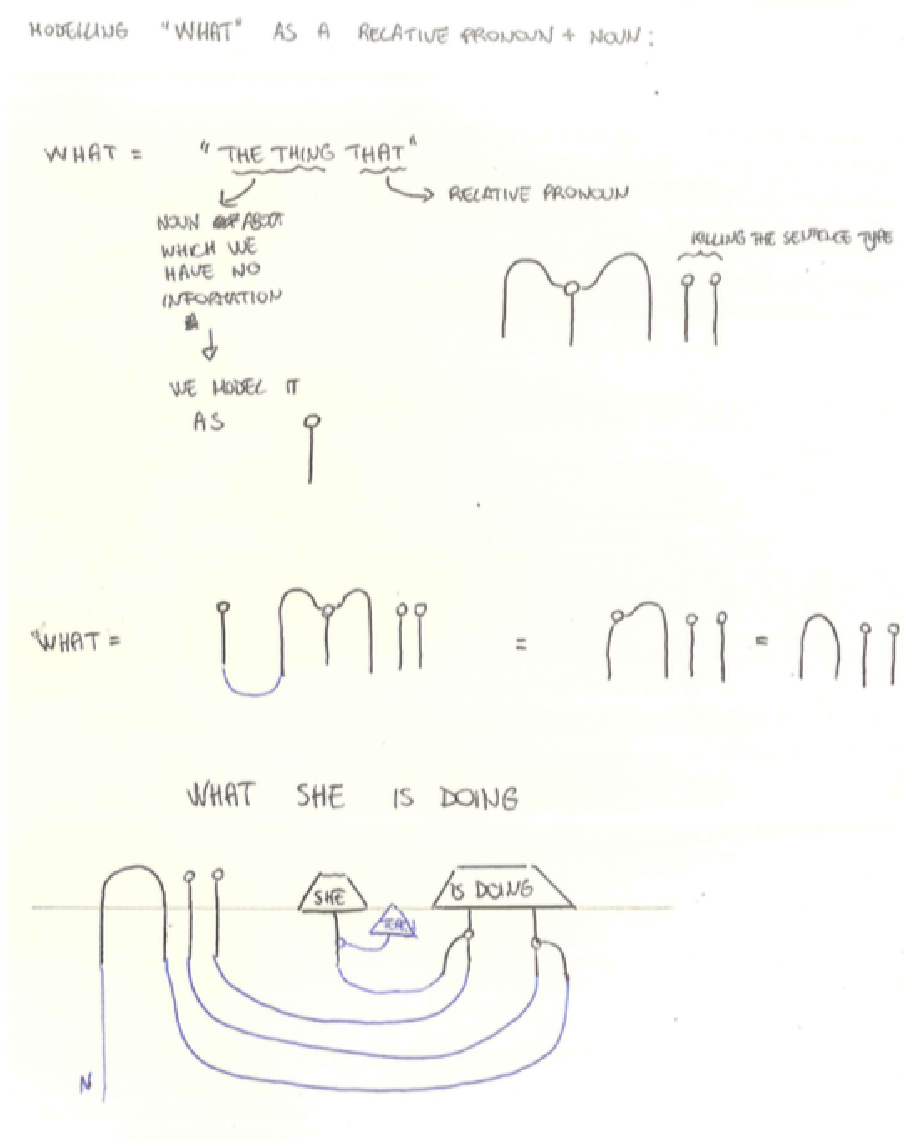
Figure 24: "Britain heard Brexit..."



Due to the presence of pronouns and adverbs, we extract the meaning of `Terry, who doesn't know what she is doing, makes a Deal` step by step, as follows.

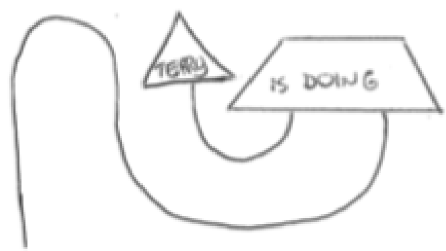
We model the adverb `what` in `What she is doing`, such that it corresponds to `the thing that`. The latter can be modelled as a no-information noun (i.e a single wired spider), followed by a relative pronoun.

Figure 25: "What she is doing"



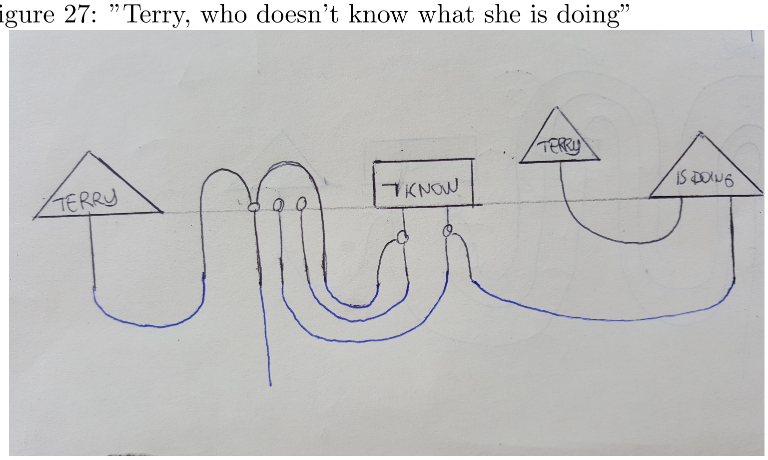
Moreover, we can notice that in Figure 25, we artificially added an extra `Terry`. This choice is an attempt to account for the presence of the reflexive pronoun `she`. Indeed, the latter could be modelled as an intersecting adjective, whose corresponding noun abstracts all the females (and thus, contains `Terry` too). Given this construction, we can artificially attach the noun to which it refers. Then, since we assumed `Terry` $\subseteq$ `she`, the intersection of `she` and `Terry` is simply `Terry`. Hence, the diagram simplifies to Figure 26.

Figure 26: "What she is doing"

copy.png 

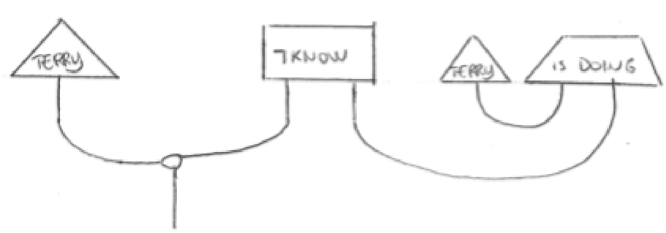
Continuing with pronouns, we have `Terry, who doesn't know what she is doing` given in Figure 27 below. Note that we replaced `does not know` with ¬`know` and we modelled the subject relative pronoun `who`, as detailed in Section 2.3.

Figure 27: "Terry, who doesn't know what she is doing"

not know.jpg 

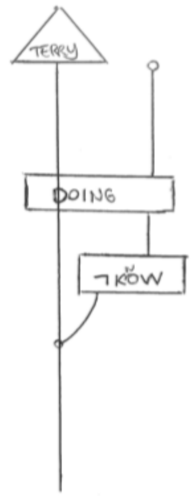
The above sentence simplifies to the following noun:

Figure 28: "Terry, who doesn't know what she is doing"



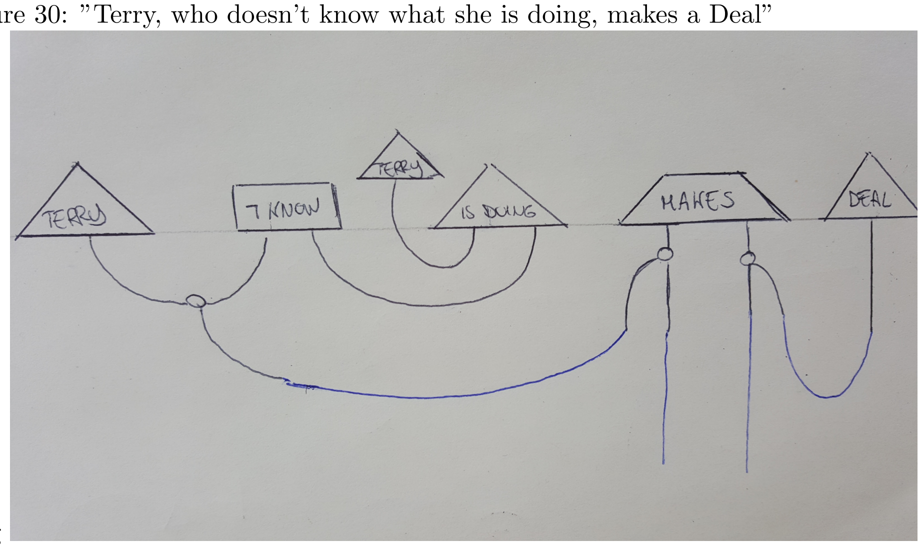
Furthermore, in Figure 29 we can observe how this sentence looks like in DisCoCirc.

Figure 29: "Terry, who doesn't know what she is doing" in DisCoCirc:



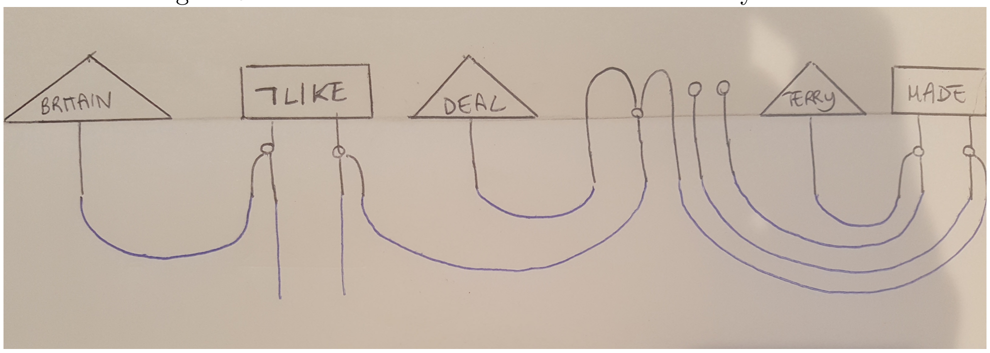
Finally, the whole sentence `Terry, who doesn't know what she is doing, makes a Deal` is given below.

Figure 30: "Terry, who doesn't know what she is doing, makes a Deal"

makes a deal.jpg 

`Britain doesn't like the Deal that Terry made` corresponds to Figure 31 below. Note the presence of the object relative pronoun `that`, and of the negative verb `¬like`, instead of `does not like`.

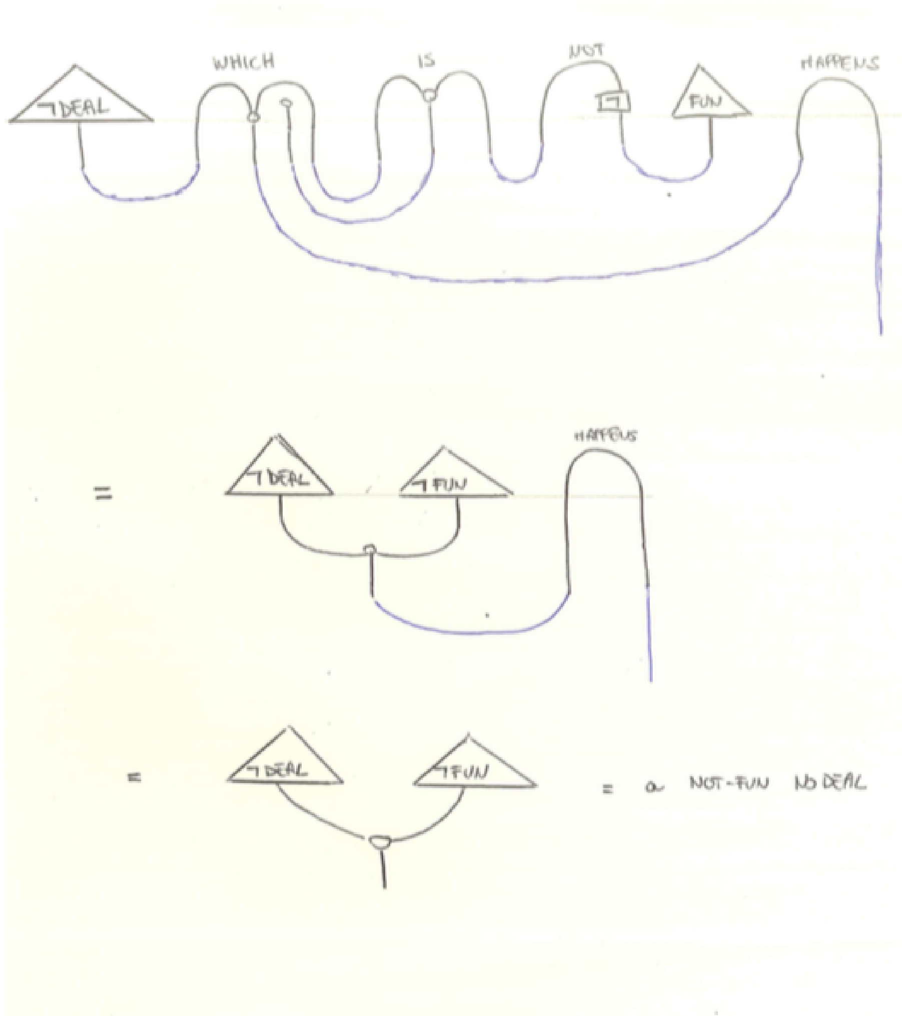
Figure 31: "Britain doesn't like the Deal that Terry made"



Our last sentence, namely `No Deal, which is not fun happens..Oops!`[2], is given below. It features the negative noun ¬`Deal`, which corresponds to `No Deal`, as seen in Figure 16. Moreover, the sentence contracts to the intersection of two nouns. This specific choice of internal wiring for `happens` was explained in Section 2.2.

[2] We omit the word `Oops`.

Figure 32: "No Deal, which is not fun, happens...Oops!"



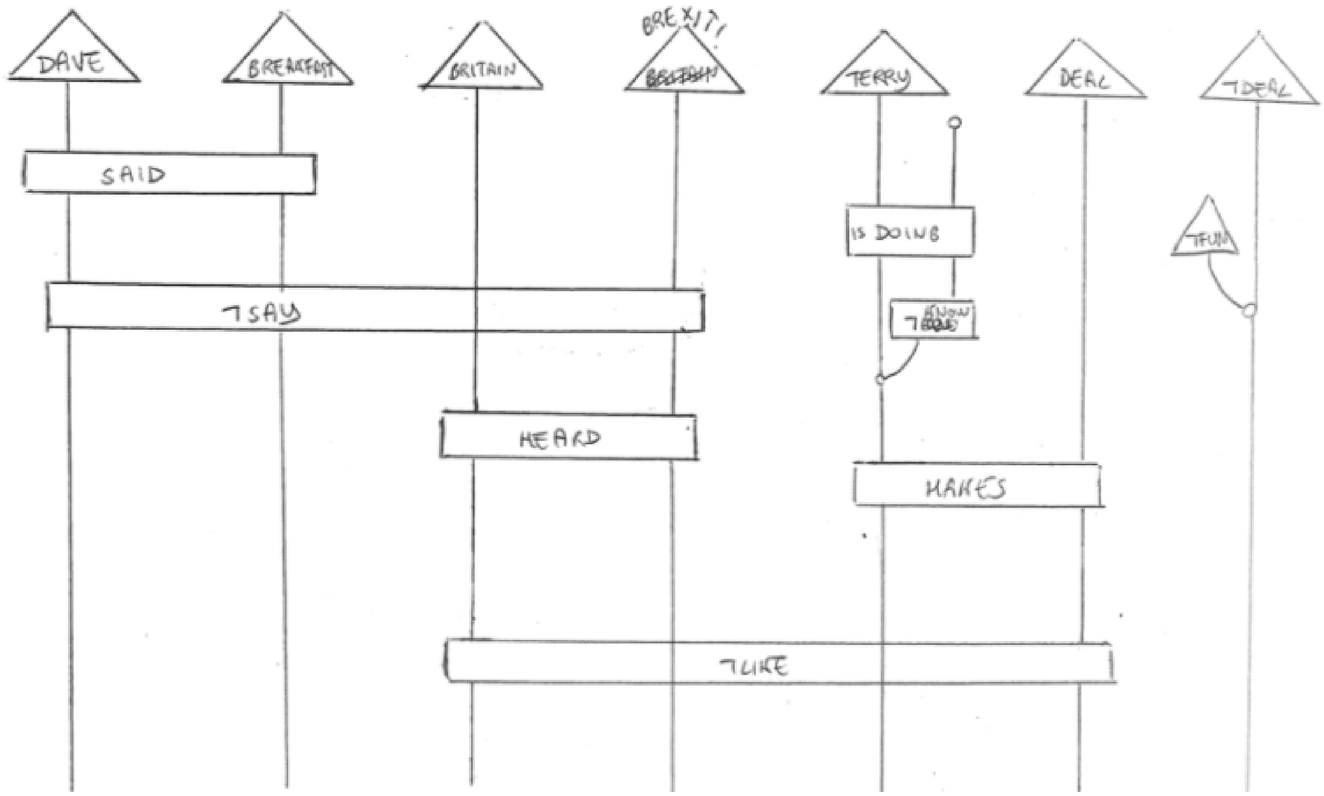
deal no fun.png

Now we move on to some DisCoCirc!

7.2 Combining sentences: the whole tale

In the previous section we have extracted the meaning of each sentence from the meaning of its words. Here, we forget about the internal structure of the sentences and we combine them together in a single DisCoCirc, as follows.

Figure 33: The whole tale in DisCoCirc:



Aside on notation: the boxes indicate sentences, which we name using their predicates. Moreover, when a wire does not interact with a certain box, we let it pass through the box.

8 Conclusion

In this project, other than having a laugh about Brexit drama, we presented a modified DisCoCat and DiscoCirc models based on the meaning category **Rel**, with noun space N. In particular, we discussed how to model nouns, adjectives, verbs, pronouns and logical function words, such that all wires are of type N.

Moreover, we constructed the noun space N as an object of **Rel**, such that it maintains some of the typical features of conceptual spaces. Following from this construction, we defined a notion of entailment and noun negation. The latter led us to define a negation operator (the magical negation box), which allowed us to define verb and sentence negation.

Furthermore, we then told a short tale about Brexit, by firstly modelling each sentence in a DisCoCat model based on the constructed meaning category, and then combining all sentences together in DisCoCirc. The meaning states were not assigned values from the noun space: this was beyond our intended scope and it is left to future development.

Finally we remind the reader that *any resemblance to real events and/or to real persons, living or dead, is purely coincidental*, and that *Brexit is coming!*

References

[1] J. Bolt, B. Coecke, F. Genovese, M. Lewis, D. Marsden, R. Piedeleu, Interacting Conceptual Spaces I: grammatical composition of concepts, *arXiv:1703.08314*, **2017**.

[2] B. Coecke, F. Genovese, M.Lewis, D. Marsden, A. Toumi, Generalised relations in cognition and linguistics, *Theoretical Computer Science 752*, 104–115, **2018**.

[3] D. Marsden, F. Genovese, Custom Hypergraph Categories via Generalised Relations, *arXiv:1703.01204 [cs.LO]*, **2017**.

[4] D. Bankova, B. Coecke, M. Lewis, D. Marsden, Graded Entailment for Compositional Distributional Semantics, *ACM*, **2016**.

[5] D. Widdows, Orthogonal Negation in Vector Spaces for Modelling Word-Meanings and Document Retrieval, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, **2003**.

[6] B. Coecke, The Mathematics of Text Structure, *to be published*, **2019**.

[7] B. Coecke, M. Sadrzadeh, S. Clark, Mathematical Foundations for a Compositional Distributional Model of Meaning, *arXiv:1003.4394 [cs.CL]* **2010**.

[8] M. Sadrzadeh, S. Clark. B. Coecke, The Frobenius anatomy of word meanings I: subject and object relative pronouns, *Journal of Logic and Computation, Special Issue: The Incomputable, an Isaac Newton Institute Workshop, 23(6), pp.1293-1317*, **2013**

[9] J. Lambek, The mathematics of sentence structure, *American Mathematical Monthly*, 65, **1958**.

[10] P. Gärdenfors, Conceptual Spaces: the geometry of thought, *The MIT Press*, **2004**.

[11] All photos and memes are downloaded from Google Images. The diagrams, on the other hand are self-made.