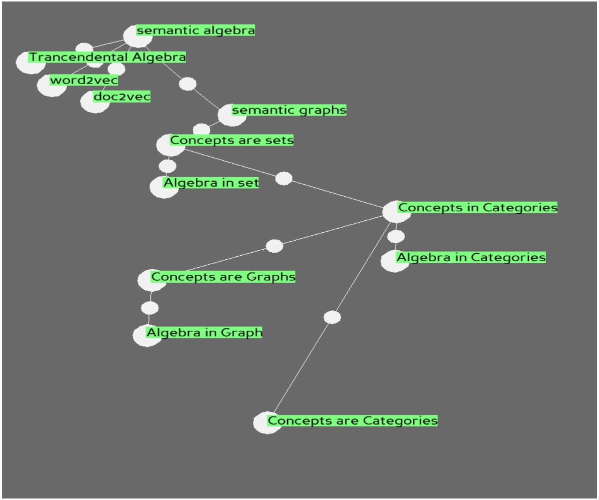
# Models for Concepts

## and Their Operations

Noah Chrein

September 18, 2019

# Ontology of Contents



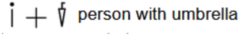

[https://github.com/nopounch/golog]

# Semantic Algebra

- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]

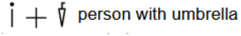
# Semantic Algebra

- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]
- The original idea is that concepts should compose in a way that is realized by algebraic operations



# Semantic Algebra

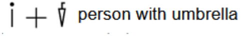
- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]
- The original idea is that concepts should compose in a way that is realized by algebraic operations

- The problem is, that this addition is ambiguous

# Semantic Algebra

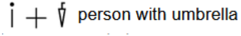
- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]
- The original idea is that concepts should compose in a way that is realized by algebraic operations




- The problem is, that this addition is ambiguous
  Does: man+ umbrella = man holding umbrella

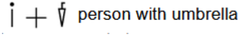
# Semantic Algebra

- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]
- The original idea is that concepts should compose in a way that is realized by algebraic operations




- The problem is, that this addition is ambiguous
  Does: man+ umbrella = man holding umbrella
  or: man+ umbrella = man, umbrella, 5 light years away

# Semantic Algebra

- Semantic Algebra can be traced back, far enough to "Trancedental Algebra" by Jacob Linzbach [1]
- The original idea is that concepts should compose in a way that is realized by algebraic operations
  
  
- The problem is, that this addition is ambiguous
  Does: man+ umbrella = man holding umbrella
  or: man+ umbrella = man, umbrella, 5 light years away
- Although eventually this failed (for exactly this reason) we have still tried to organize concepts algebraicly

# Word2Vec

- Modern Computational capabilities have allowed for more complicated approaches to semantic algebra.

# Word2Vec

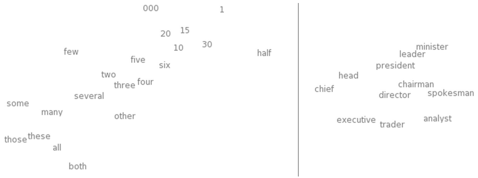
- Modern Computational capabilities have allowed for more complicated approaches to semantic algebra.
- in 2013, a paper by Mikolov detailed a way to organize words into a high dimensional real space using neural network.

# Word2Vec

- Modern Computational capabilities have allowed for more complicated approaches to semantic algebra.
- in 2013, a paper by Mikolov detailed a way to organize words into a high dimensional real space using neural network.
- This network's task was to find an "Word Embedding" W:Words$\rightarrow \mathbb{R}^n$ [Mikolov][2]

# Word2Vec

- Modern Computational capabilities have allowed for more complicated approaches to semantic algebra.
- in 2013, a paper by Mikolov detailed a way to organize words into a high dimensional real space using neural network.
- This network's task was to find an "Word Embedding" W:Words$\rightarrow \mathbb{R}^n$ [Mikolov][2]
- The result is a word embedding that places contextually relevant words close to each other.




graphics from C. Olah[3]

# Magical Analogies

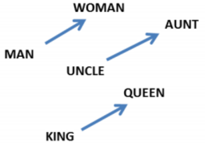
- A side effect of this word embedding, is that vector operations seemed to represent analogies

# Magical Analogies

- A side effect of this word embedding, is that vector operations seemed to represent analogies
- if you run vector operations, and you would get something like:

$$W("Woman") - W("Man") \simeq W("Queen") - W("King")$$

" Man is to Woman as King is to Queen"




graphics from C. Olah[3]

# Contextual Analogies

- These analogies are not magic

# Contextual Analogies

- These analogies are not magic
- they are built by the context of sentences in the training data.

# Contextual Analogies

- These analogies are not magic
- they are built by the context of sentences in the training data.
- Sentences like "The wall is red" and "the wall is blue" allow us to "swap" the words blue and red

# Contextual Analogies

- These analogies are not magic
- they are built by the context of sentences in the training data.
- Sentences like "The wall is red" and "the wall is blue" allow us to "swap" the words blue and red
- this is the meaning behind a vector operation like W("blue")-W("red")

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic,

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic, to Sartre,

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic, to Sartre, you might end up with man + teacup =

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic, to Sartre, you might end up with
  man + teacup =
  "Man holding a teacup"

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic, to Sartre, you might end up with  
  man + teacup =  
  "Man holding a teacup"  
  "Man swimming in a teacup"

# Contextual Analogies

- There are more sophisticated papers (such as doc2vec[4]) where we can contruct sentence embeddings
- something like "man + teacup" might give you a vector close to the sentence "man holding a teacup".
- But this is ambiguous and highly dependent on the training data.
- depending on your training data, from British novelists, Alice in Wonderland fanfic, to Sartre, you might end up with
  man + teacup =
  "Man holding a teacup"
  "Man swimming in a teacup"
  "A man and a teacup exist"

# Concepts have internal Data

- This ambiguity is exactly the issue...

# Concepts have internal Data

- This ambiguity is exactly the issue...
- Can we define an unambiguous "addition" that makes semantic sense?

# Concepts have internal Data

- This ambiguity is exactly the issue...
- Can we define an unambiguous "addition" that makes semantic sense?
- One that, If still generated by contextual biases, at least allows us to understand those biases

# Concepts have internal Data

- This ambiguity is exactly the issue...
- Can we define an unambiguous "addition" that makes semantic sense?
- One that, If still generated by contextual biases, at least allows us to understand those biases
- We already have a model of how concepts relate to other concepts contextually

# Concepts have internal Data

- This ambiguity is exactly the issue...
- Can we define an unambiguous "addition" that makes semantic sense?
- One that, If still generated by contextual biases, at least allows us to understand those biases
- We already have a model of how concepts relate to other concepts contextually
- What we lack is a model for a concept, *internally*

# Concepts have internal Data

- This ambiguity is exactly the issue...
- Can we define an unambiguous "addition" that makes semantic sense?
- One that, If still generated by contextual biases, at least allows us to understand those biases
- We already have a model of how concepts relate to other concepts contextually
- What we lack is a model for a concept, *internally*
- To differentiate between "Man holding a teacup" and "Man swimming in a teacup", we should consult the internals of the sum Man + Teacup

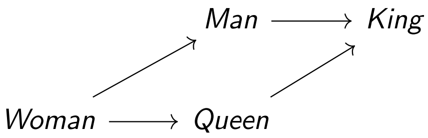
# Concepts are Objects

I think we already have it

I am talking about category theory

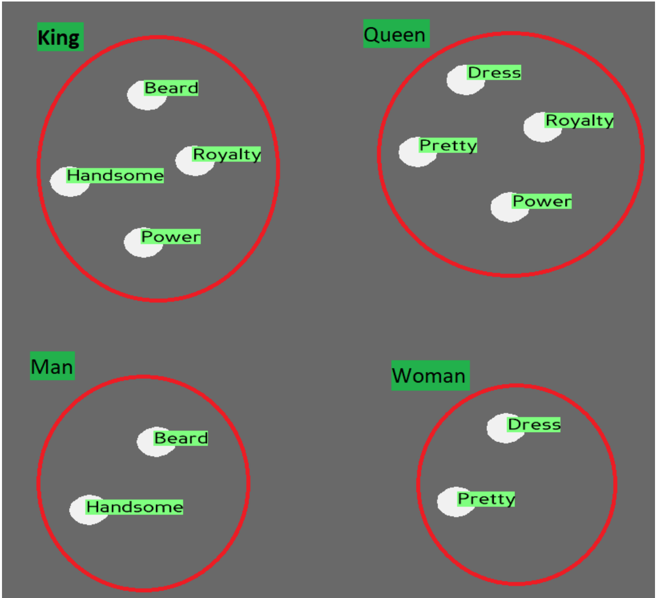
# Concepts are Sets?

- the goal is to create unambiguous conceptual addition by consulting the internals of a concept
- Our universe for concepts was a vector space, but let's now consider this as a (directed) graph




- from this perspective, we can see the nodes clearly
- let's give some data to these nodes
- assume each node is instead a set

# Man, Woman, King, Queen as Sets



# Algebraic Operations on Sets

- define set "addition" to be the union

# Algebraic Operations on Sets

- define set "addition" to be the union
- define set "subtraction" to be excision

# Algebraic Operations on Sets

- define set "addition" to be the union
- define set "subtraction" to be excision
- denote the set associated to a concept A by O(A)
  then:

# Algebraic Operations on Sets

- define set "addition" to be the union
- define set "subtraction" to be excision
- denote the set associated to a concept A by O(A)
  then:
  $O(\text{King}) = \{\text{'beard', 'power','handsome','royalty'}\}$

# Algebraic Operations on Sets

- define set "addition" to be the union
- define set "subtraction" to be excision
- denote the set associated to a concept A by O(A) then:
  O(King) = {'beard', 'power','handsome','royalty'}
  O(King) - O(Man) = {'power','royalty'}

# Algebraic Operations on Sets

- define set "addition" to be the union
- define set "subtraction" to be excision
- denote the set associated to a concept A by O(A)
  then:
  O(King) = {'beard', 'power','handsome','royalty'}
  O(King) - O(Man) = {'power','royalty'}
  O(King) - O(Man) + O(Woman) =
  {'dress', 'power','pretty','royalty'} = O(Queen)

# Sets aren't the "best" model

- We can describe a "universe with a man and a teacup" using disjoint union:

  $$O(\text{Universe with man and teacup}) = O(\text{Man}) \coprod O(\text{teacup})$$

# Sets aren't the "best" model

- We can describe a "universe with a man and a teacup" using disjoint union:

  $$O(\text{Universe with man and teacup}) = O(\text{Man}) \coprod O(\text{teacup})$$

- but we lack the expressive power to describe a "man holding a teacup" as some operation

# Sets aren't the "best" model

- We can describe a "universe with a man and a teacup" using disjoint union:

  $$O(\text{Universe with man and teacup}) = O(\text{Man}) \coprod O(\text{teacup})$$

- but we lack the expressive power to describe a "man holding a teacup" as some operation
- the conceptual sets of "man" and "teacup" do not share any underlying concepts (like king and queen did) so their union will always be disjoint

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition
- we can go on an on choosing better and better models:

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition
- we can go on an on choosing better and better models:Simplicial Sets,

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition
- we can go on an on choosing better and better models:Simplicial Sets, Higher Simplicial Sets,

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition
- we can go on an on choosing better and better models:Simplicial Sets, Higher Simplicial Sets, Topological Spaces, Vector Spaces, Hilbert Spaces ...

# Sets aren't the "best" model

- Sets lack the expressive power for internal relationships
- we may be tempted to choose graphs
- but even these lack power: no composition
- we can go on an on choosing better and better models:Simplicial Sets, Higher Simplicial Sets, Topological Spaces, Vector Spaces, Hilbert Spaces ...
- before **choosing** a "best" model to describe the internals let's agree on how to work with any model

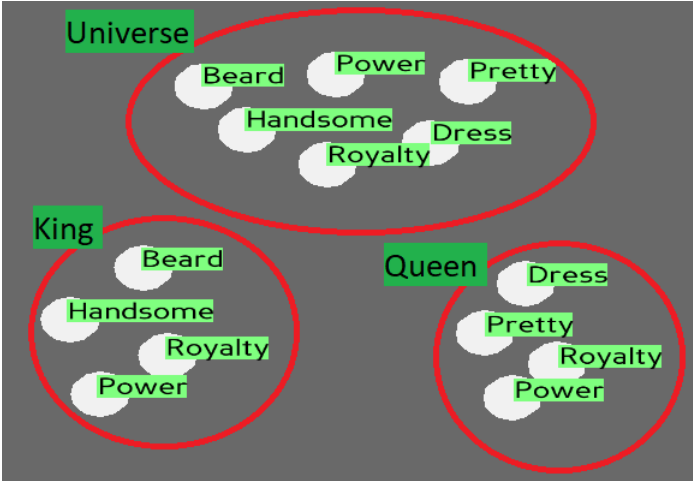
# Lifting to arbitrary objects

- Let's just assume that whatever models we have, they live inside of some category $\mathfrak{C}$
- We can recast our union and excision in categorical semantics alone
- This concept is known as a Universal Property

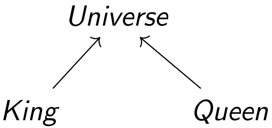
# Some necessary assumptions

1) Our concept's internals are objects of some category
2) There is some Universe object (to compare concepts)
3) (For the categorically minded) $\mathfrak{C}$ must also be co/complete with a terminal and initial object

# Universe Example

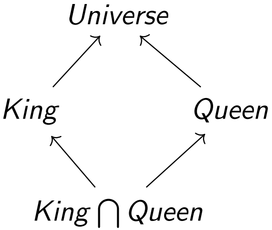


# Intersection is Pullback



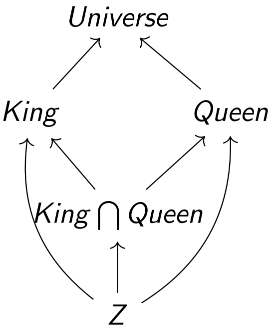

begin with our embeddings
into our universe set

# Intersection is Pullback



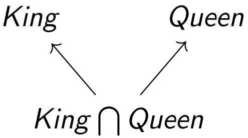

begin with our embeddings
into our universe set
The intersection is something
that embeds into both concepts

# Intersection is Pullback



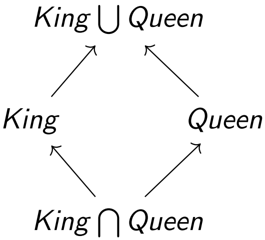

begin with our embeddings
into our universe set
The intersection is something
that embeds into both concepts
And it is the biggest thing
that embeds into both concets

# Union is Pushout



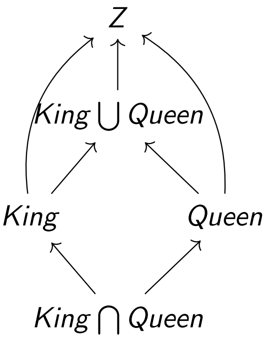

Now that we have our intersection

# Union is Pushout



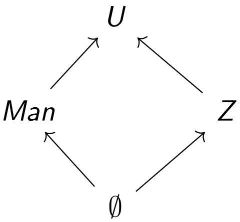

Now that we have our intersection
Both concepts embed into the union
Such that it's intersection goes
to the same place

# Union is Pushout



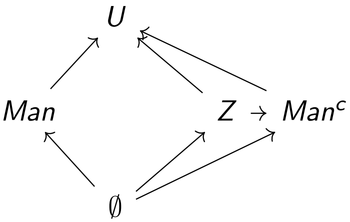

Now that we have our intersection
Both concepts embed into the union
Such that it's intersection goes
to the same place
The Union is the smallest
thing that does this

# Compliment and set subtraction



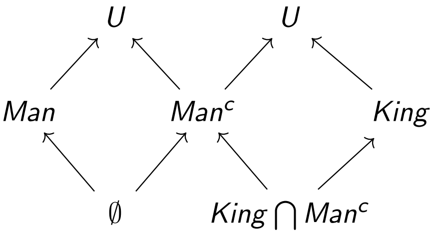

Assume that some concept and another have empty intersection

# Compliment and set subtraction



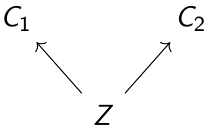

The "Compliment" of that concept is the largest thing with empty intersection

# Compliment and set subtraction



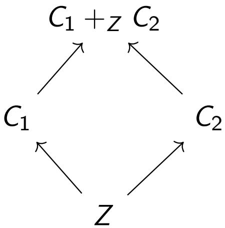

We can then combine the intersection and the compliment to for the subtraction $King - Man = King \bigcap Man^c$

# Gluing



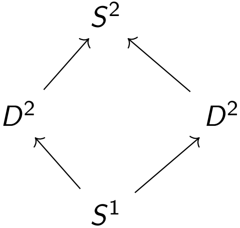

Without the "Universe" Object to compare the internals of two concepts, we can still "force" two objects together by asserting their intersection

# Gluing




Using a universality condition, we can find a "best" concept to complete the diagram

# Gluing




The Classic example is gluing two disks together by their boundary circle to get a sphere

# Summary so far

- We Came up with a case for describing the internals of a concept

# Summary so far

- We Came up with a case for describing the internals of a concept
- We tried this with sets, discussed some algebra of sets

# Summary so far

- We Came up with a case for describing the internals of a concept
- We tried this with sets, discussed some algebra of sets
- We found some downside with the expressability of sets

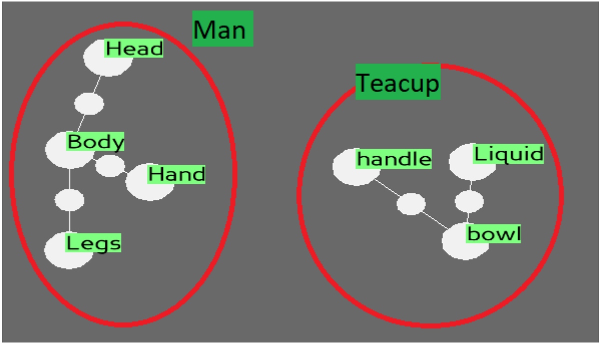
# Summary so far

- We Came up with a case for describing the internals of a concept
- We tried this with sets, discussed some algebra of sets
- We found some downside with the expressability of sets
- We generalized our operations from sets to arbitrary "nice" categories

  Now let's talk about some specific categories

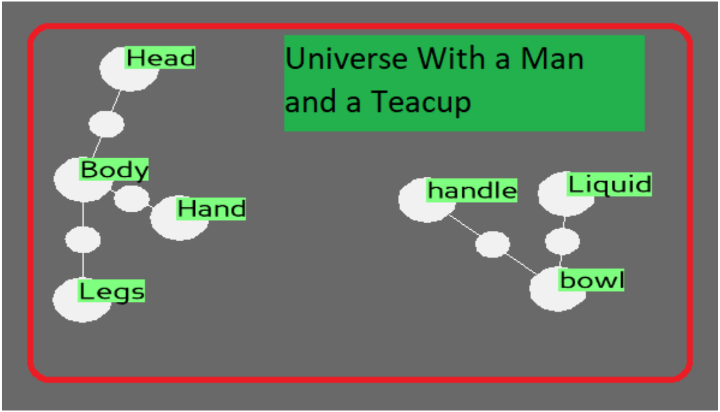
# Concepts as Graphs

- so now the question is, can we use our conceptual algebra to differentiate the possible man + teacup compositions
- Lets lift our internal conceptual representations from Set to Graph
  So for example, man and teacup:



# Graph Disjoint Union

As it is, we can still define the disjoint union of these two graphs. This will give us a "Universe With an Man and a Teacup":



# Relations as Graphs

Let's define two 1-edge graphs that will represent "Holds" and "Swims"

# Relations as Graphs

Let's define two 1-edge graphs that will represent "Holds" and "Swims"

$H = \{supporter \stackrel{holds}{\rightarrow} supported\}$

# Relations as Graphs

Let's define two 1-edge graphs that will represent "Holds" and "Swims"

$H = \{supporter \stackrel{holds}{\rightarrow} supported\}$

$S = \{Under \stackrel{swims}{\rightarrow} Liquid\}$

# Relations as Graphs

Let's define two 1-edge graphs that will represent "Holds" and "Swims"

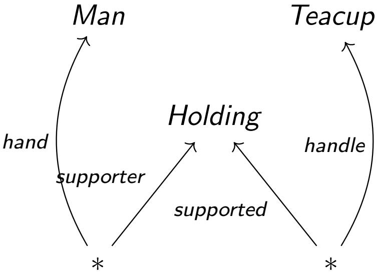
$H = \{supporter \stackrel{holds}{\rightarrow} supported\}$

$S = \{Under \stackrel{swims}{\rightarrow} Liquid\}$

And the one point graph with no edge *

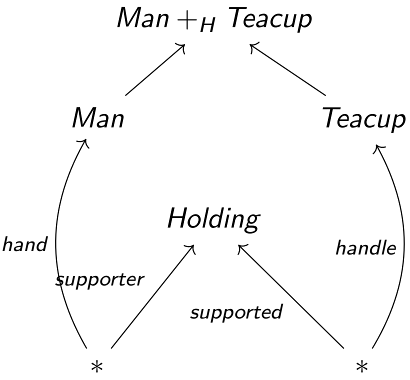
$* = \{\bullet\}$

# Man holding teacup



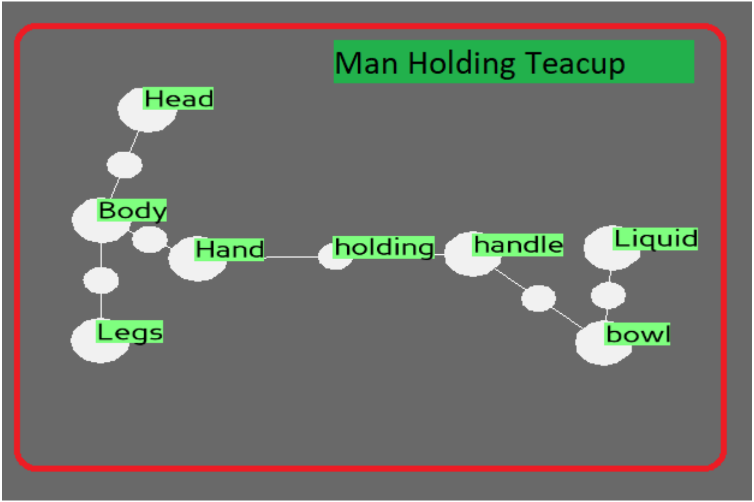

Consider the diagram H
a subcategory of graph
The shown functions send
the single point •
to the name of the function.
e.g. hand(•) = hand

# Man holding teacup



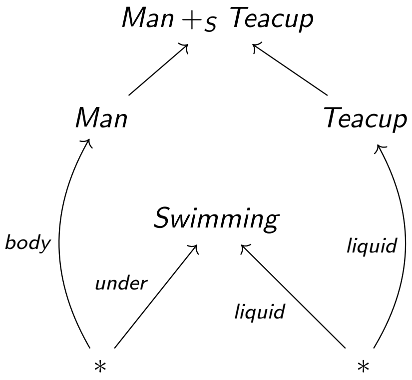

By gluing H (Universality)
We can construct a conceptual addition Man $+_S$ Teacup

# Man Holding Teacup

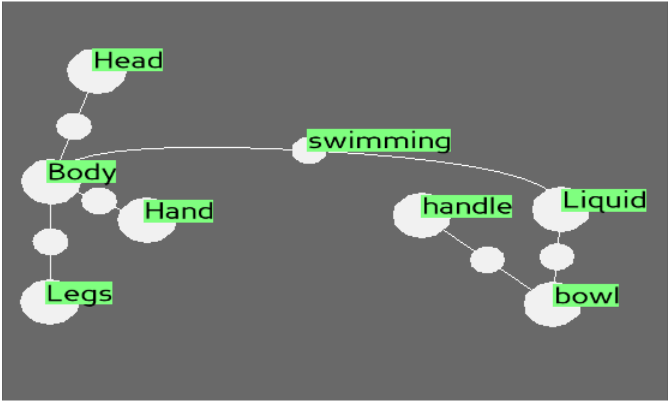


# conceptual addition of man swimming in teacup

Of course, now we can do this for "swimming" as well

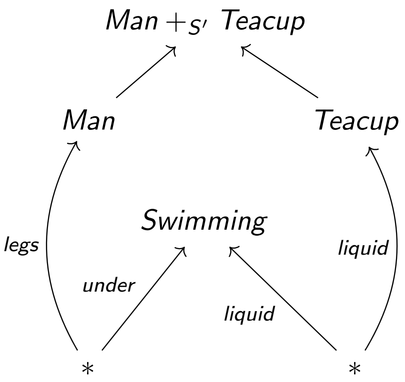


# Man Swimming in Teacup

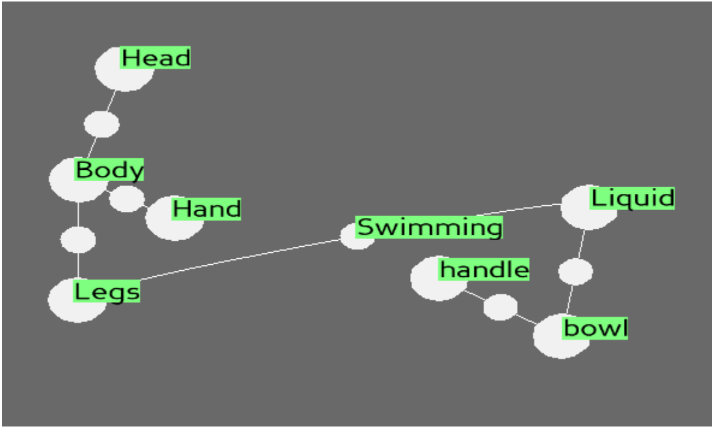


# Conceptual addition of Man wading in Teacup

Of course, now we can do this for "swimming" as well

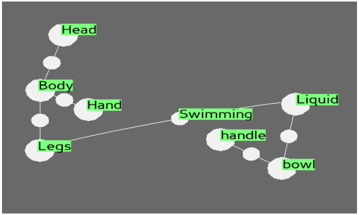


# Man Wading in Teacup



# Graphs: Externally and Internally

- but remember that our original "concept space" is also a graph
- on the outside I might have a relation like:
  man $\xrightarrow{\text{wading in}}$ teacup
- on the inside it looks something like this:



# Graphs: Externally and Internally

- on the inside I have the categorical semantics to define

$$\text{"man} +_S \text{teacup"}$$

  but on the outside I do not...
- If, on the outside, I had some categorical structure I might be able to compare universal properties

# Concepts are Categories(?)

- realizing the inside of concepts in some category, we can construct universal properties

# Concepts are Categories(?)

- realizing the inside of concepts in some category, we can construct universal properties
- if the outside "collection of concepts" formed a category as well, we could derive universal properties before expanding

# Concepts are Categories(?)

- realizing the inside of concepts in some category, we can construct universal properties
- if the outside "collection of concepts" formed a category as well, we could derive universal properties before expanding
- upon expansion we will better realize what was meant, but before then, we should still be able to guess

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"
- That is, both the external collection of concepts, and internals of particular concept, should be categories

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"
- That is, both the external collection of concepts, and internals of particular concept, should be categories
- this way, the inside universal property and the outside Universal property "guess" can agree up to some functorality condition

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"
- That is, both the external collection of concepts, and internals of particular concept, should be categories
- this way, the inside universal property and the outside Universal property "guess" can agree up to some functorality condition
- further, the elements on the inside of a concept can be expanded to provide more details

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"
- That is, both the external collection of concepts, and internals of particular concept, should be categories
- this way, the inside universal property and the outside Universal property "guess" can agree up to some functorality condition
- further, the elements on the inside of a concept can be expanded to provide more details
  (This is the role of ontological expansion)

# Categories all the way down

- If both the inside and outside are objects in the same category, then we can do multiple "expansions"
- That is, both the external collection of concepts, and internals of particular concept, should be categories
- this way, the inside universal property and the outside Universal property "guess" can agree up to some functorality condition
- further, the elements on the inside of a concept can be expanded to provide more details
  (This is the role of ontological expansion)
  (For another talk)

# References

[1] "Trancendental Algebra" http://keyboardfire.com/s/transalg/
[2] "Distributed Representations of Words and Phrasesand their Compositionality"
https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf
[3] "Deep Learning, NLP, and Representations"
https://colah.github.io/posts/2014-07-NLP-RNNs-Representations/
[4] "Distributed Representations of Sentences and Documents"
https://arxiv.org/abs/1405.4053
[5] "Category Theory For Scientists"
http://math.mit.edu/ dspivak/teaching/sp13/CT4S.pdf