Using Grounded Word Representations to Study Theories of Lexical Concepts

Anonymous NAACL submission

Abstract

The fields of cognitive science and philosophy have proposed many different theories for how humans represent "concepts". Multiple such theories are compatible with state-of-the-art NLP methods, and could in principle be operationalized using neural networks. We focus on two particularly prominent theories–Classical Theory and Prototype Theory–in the context of visually-grounded lexical representations. We compare when and how the behavior of models based on these theories differs in terms of categorization and entailment tasks. Our preliminary results suggest that Classical-based representations perform better for entailment and Prototype-based representations perform better for categorization. We discuss plans for additional experiments needed to confirm these initial observations.

1 Introduction

There are many theories and proposed definitions for what exactly constitutes a "concept". Which definition is the right one is a hotly debated topic in philosophy and psychology, which has involved a wide range of in-principle as well as empirical arguments (Laurence and Margolis, 1999). Despite the lack of consensus as to their definition, it's generally agreed that representations of concepts play a key role in natural language understanding, as the meaning of natural language expressions are necessarily defined in terms of their denotations–i.e. the aspects of the grounded (non-linguistic) world to which the expression refers. For example, reasoning about how the word *"owl"* relates to the word *"bird"* requires consideration of how *the thing or things referred to by "owl"* relates to *the thing or things referred to by "bird"*. Thus, representations of the concepts to which language refers is a key part of general language understanding.

It is not obvious, however, how one should chose to represent concepts computationally, especially given that current state-of-the-art neural models of grounded language can be seen as compatible with a number of theories for concepts, depending on how the architectures and algorithms are constructed. Thus, in this paper, we focus in particular on lexical concepts, and study two prominent theories which have both wide support–as well as substantial criticism–within the psychology and philosophy communities (Laurence and Margolis, 1999). The first, Classical Theory, represents concepts as the set of necessary-and-sufficient conditions which define the extension of the concept. For example, the representation of *owl* is the set of conditions such that, if and only if some entity meets every condition, that entity is an *owl*. Classical Theory is the most frequently cited in linguistics and NLP– it is the theory underlying traditional formal semantics–and is often formalized in terms of set theory, i.e. the extension of *"owl"* is the set of all owls. The second theory we explore is Prototype Theory, which represents concepts as a single, prototypical instance of that concept. For example, the representation of *owl* would be a particular instance of owl that captures the most characteristic, salient, typical, or otherwise important properties associated with owls. The degree to which some entity falls within the extension of *owl* is then a function of how "similar" that entity is to the prototype of owl. Thus, unlike Classical Theory, there is no clear notion of what is required in order to be an owl, and an entity may be judged to be an owl on the basis of "resemblance" despite having few definable properties in common with the prototype.

There are many points of differentiation that one might make between Classical Theory and Prototype Theory. In particular, Classical Theory is typically associated with discreteness and binary-ness (e.g. an entity either is an owl or it is not) while Prototype Theory is associated with graded judgements. By this distinction, it seems that Classical Theory is at odds with the state-of-the-art in NLP, which hinges on continuous representations and probabilistic judgements. However, in this paper we highlight a different distinction between Classical and Prototype Theory, which enables both theories to be operationalized in terms of continuous representations. Specifically, we frame Classical Theory as concerned primarily with representing *boundaries between classes* and Prototype Theory as concerned primarily with representing the *centers of classes*. That is, Classical Theory strives to determine the line that separates the least owl-like owl from most owl-like non-owl, while Prototype Theory strives to determine the properties that are most likely true of owls in general.

We conduct an empirical comparison of these two theories by providing computational instantiations of each in the context of visually-grounded word representations. Specifically, we uses images with a given label (i.e. images of owls) to represent observed instances of each concept, and encode all images into a shared space using a Variational Autoencoder (VAE). We then build a Classical-based representation by computing the boundary which encompasses all instances of a given concept, and build a Prototype-basesd representation by computing the center of mass among all instances of a given concept. We compare these two models in terms of their performance on two tasks: 1) categorization (i.e. determining whether an instance falls within the extension of the concept) and 2) entailment (deciding whether one concept subsumes another). Our initial results suggest that the Classical-based representation consistently outperforms the Prototype-based representation on tasks related to entailment, even when we take into account the gradability of human entailment judgments. However, our results also suggest that the Prototype-based representation is better suited to perform the categorization task, although further investigation is needed to draw a complete comparison.

2 Definitions

2.1 Notation

We will use C to represent a concept and x to represent a potential "instance" of the concept. Intuitively, we can think of x as an entity when C is a concept corresponding to a noun like *"cat"*, but x might also be an event, property, or any other more abstract possible referent which might be considered to fall within the extension of C. $\mathcal{C}$ and $\mathcal{X}$ represent the space of concepts and of instances, respectively. We assume that a representation of a concept must support the tasks of categorization and entailment, as follows:

Categorization: A function $f_C : \mathcal{X} \rightarrow [0,1]$ which returns the probability that x is an falls within the extension of C.

Entailment: A function $\mathit{entail} : \mathcal{C} \times \mathcal{C} \rightarrow [0,1]$ which returns the probability that $C2$ can be inferred from $C1$.

2.2 Classical Theory

In Classical Theory, a concept is represented as a set of conditions which are necessary and sufficient in order for an entity to fall within the extension of the concept. Typically, in formal linguistics, this is discussed in terms of set theory: i.e. the denotation of a word is the set of instances in $[\![C]\!] \subseteq \mathcal{X}$ which forms the extension of that word. Thus, f_C is simply the characteristic function of this set. As classical theory is primarily concerned with defining clear boundaries between what can and can not be considered a member of the concept, this is best captured as a binary function (instances either are in the set or they are not):

$$f_C(x) = \begin{cases} 1 \text{ if} & x \in [\![C]\!] \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Then, C_1 is said to entail C_2 if $[\![C_1]\!] \subseteq [\![C_2]\!]$:

$$\mathit{entail}(C_1, C_2) = \begin{cases} 1 \text{ if} & \forall x (f_{C_1}(x) \leq f_{C_2}(x)) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

That is, whenever $f_{C_1}(x) = 1$, we must also have $f_{C_2}(x) = 1$. We also can consider a relaxed definition that supports graded (probabilistic) judgments of entailment. Specifically, we can say that the degree to which $C1$ entails $C2$ is determined by the degree of overlap between these sets:

$$\mathit{entail}(C_1, C_2) = \frac{\sum_{x \in \mathcal{X}} f_{C1}(x) \times f_{C2}(x)}{\sum_{x \in \mathcal{X}} f_{C1}(x)} \quad (3)$$

That is, the probability that $C1$ entails $C2$ is exactly the probability that a given instances of $C1$ is also an instance of $C2$.

2.3 Prototype Theory

In Prototype Theory, a concept is represented as a single "prototype"–i.e. an instance that falls within the extension of the concept and captures the most relevant, salient, or important properties of the concept. In contrast to Classical Theory, the features of the prototype do not represent necessary criteria–it is possible for an instance to fall within the extension of the concept despite having few features in common with the prototype. Concepts, then, are represented as a tuple containing an exemplar x_C and a distance function $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ which specifies how similar an arbitrary instance is to the exemplar. While there is no crisp definition of the extension of the concept, it is generally accepted that the criteria for inclusion in the extension must be proportional to the distance function (Osherson and Smith, 1981; Kamp and Partee, 1995):

$$f_C(x) \propto d(x, x_C) \tag{4}$$

That is, for any pair of instances x, y, if $d(x, x_C) < d(y, x_C)$, it cannot be the case that y is in the extension of C but x is not.

Traditional descriptions of Prototype Theory–i.e. those described in Rosch and Lloyd (1978); Kamp and Partee (1995)–do not explicitly define how to reason about entailment under Prototype Theory. Osherson and Smith (1981) proposed the use of fuzzy set theory (Zadeh et al., 1996) as a means for incorporating Prototype Theory within the familiar logical framework for reasoning about entailment. However, this approach has received significant criticism regarding the predictions it makes about compositionality (Osherson and Smith, 1981). Thus, we consider an alternative, simple definition of entailment which simply says that C_1 entails C_2 to the extent that the exemplar of C_1 falls within the extension of C_2:

$$entail(C_1, C_2) = f_{C_2}(x_{C_1}) \tag{5}$$

We begin with this definition as it is straightforward and reflects the basic spirit of Prototype Theory, without forcing it to look like set theory. We will consider alternative definitions in future work.

3 Instantiation

We focus on lexical concepts, specifically those corresponding to common nouns. We instantiate the definitions given in Section 3.1 using images to represent "instances". That is, our $\mathcal{X}$ is the space of all images and our $\mathcal{C}$ maps one-to-one onto English nouns. A similar approach, using images as a representation of "the world", has been used previously (Young et al., 2014). We adopt this approach as it enables a fairly direct way to instantiate abstract formal theories using representations (pixels) which can be handled straightforwardly by current computational models. We do not make the claim that visual attributes are the only relevant attributes which factor into representations of concepts. Rather, our focus is on testing in general how the choice of representation affects the predictions made by models, assuming that some representation of "the world" is given *a priori*. In other words, our choice to use only visual attributes is a methodologically-motivated choice, not a theoretically-motivated one.

3.1 Models

VAE. We encode all of our images into a shared space using a standard variational autoencoder (VAE) (Kingma and Welling, 2013). An advantage of using a VAE in this research is that latent features are encouraged to match a normal distribution, enforcing a structure on the latent space that allows euclidean geometric manipulations such as interpolation. This allows us to instantiate simple and intuitive euclidean evaluations when comparing theories. We train a VAE to reconstruct image encodings from a pretained CNN. In the following descriptions, $\vec{x} = VAE(CNN(x))$, i.e. the d-dimensional encoding of an image obtained by applying a pertained image classifier followed by our VAE encoder.

Classical-Based Method. Our definition of Classical Theory requires only that we can define the boundary for each concept. Given a set $\mathcal{X}_C$ of instances of a concept C–i.e. the set of images x observed with label C–we define this boundary to be the convex hull $\mathcal{H}_C$ computed over $\vec{x}$ for every $x \in \mathcal{X}_C$. That is, we compute the literal boundary surrounding a set of encoded instances (shown as solid lines in Figure 1). We can then evaluate whether an arbitrary new instance x is a member of C by computing whether $\vec{x}$ falls within this boundary (Eq. 6). We can then produce entailment judgments using Eq. 2 or 3 exactly.

$$f_C(x) = \begin{cases} 1 \text{ if} & \vec{x} \cdot \mathcal{H}_C \leq 0 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

When evaluating on the entailment tasks (Section 5.2) we consider two variants of this Classical-based representation. First, we consider a “strict” interpretation of entailment, where $C_1 \rightarrow C_2$ iff *every* instance in C_1 is also in C_2. Second, we consider a soft representation in which which $C_1 \rightarrow C_2$ if the proportion instances from C_2 that are in C_1 (i.e. Eq. 3) is at least τ. When we use this soft representation, we set τ using performance on a held-out validation set.

Prototype-Based Method. Our definition of Prototype Theory requires that we can define a prototype instance and a distance function for each concept. Again, given $\mathcal{X}_C$, the set of images x observed with label C, we approximate a probability density function ϕ_C - in this case, as a multivariate normal distribution. We then define the prototype $\vec{x_C}$ to be the mode of ϕ_C. The distance function d can then be defined as:

$$d(\vec{x}, \vec{x_C}) = \frac{\phi_C(\vec{x})}{\phi_C(\vec{x_C})} \tag{7}$$

or the density at point $\vec{x}$ in ϕ_C. Because the density may evaluate to a value greater than 1, we normalize by density at the prototype $\phi_C(\vec{x_C})$. This results in values in the range $[0, 1]$, which are more interpretable and comparable across scenarios. We parameterize density function ϕ_C as a multivariate normal distribution with mean μ_C and covariance σ^2_C, resulting in prototype $\vec{x_C} = \mu_C$. We chose this distance function as it is arguably the simplest way to compute “distance to the prototype” which still allows asymmetry. That is, pure euclidean distance would be simpler, but would lose the ability to represent directionality, meaning e.g. *“owl”* would be as prototypical of *“bird”* as *“bird”* is of *“owl”*. In future work, we will consider different definitions of prototype and/or more complex distance functions, as well as alternative, i.e. non-Gaussian, representations.

When evaluating on the categorization tasks (Section 5.2), we must use this distance function to make a binary decision about whether or not an instance falls within the extension of the concept. Thus, analagous to how we softened the Classical-based representation, which stricten our Prototype-based representation by defining threshold τ, and saying that $f_C(x) = 1$ *iff* $d(\vec{x}, \vec{x_C}) \leq \tau$. Again, when used, we set τ empirically based on performance on a validation set.

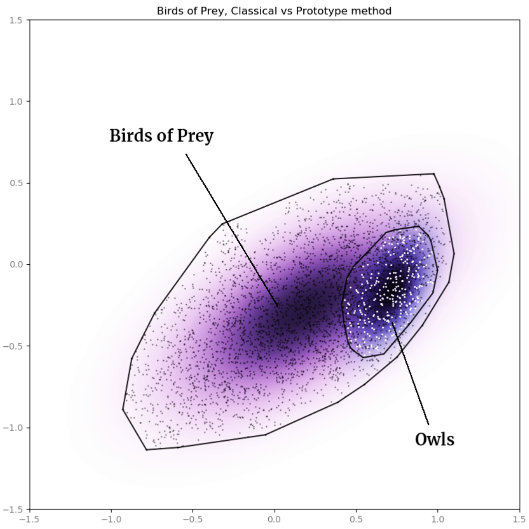


Figure 1: Encodings of *“bird_of_prey”* and *“owl”* as black and white dots respectively. The convex hull (Classical-based representation) is represented by the black lines. Colored gradients show multivariate normal distributions (Prototype-based representation).

3.2 Training

We train our VAE on IMAGENET (Deng et al., 2009), which consists of approximately 1,000 images for each of 1,000 fine-grained, mutually exclusive categories corresponding to common nouns/noun phrases (e.g. *“great_grey_owl”*, *“knee_pad”*). These class labels have been mapped onto the WORDNET (Miller et al., 1990) ontology, which provides a tree structure of hypernym-hyponym relationships. Since we want representations for both fine-grained concepts as well as higher-level concepts (in order to evaluate entailment), we compose data of high-level concepts from their lower-level hyponyms. For example, for the high-level concept *“bird_of_prey”*, we take all hyponyms of *“bird_of_prey”* according to WORDNET (e.g. *“great_grey_owl”*, *“kite”*). Of these, we identify those in IMAGENET and gather instances of these subclasses to comprise the set for the superclass *“bird_of_prey”*.

We hold out 100 instances of each low-level class to keep for testing. We split our data evenly between hypernym/hyponym labels and between train/test sets to ensure that, e.g., if a particular image of an owl is used as a *“bird_of_prey”* during training, then that same instance is not seen as an *“owl”* nor as a *“bird_of_prey”* during test. The same image might been see as both and *“owl”* and a *“bird_of_prey”* during training.

We feed each image through a pretrained image classifier (Inception v3) (Szegedy et al., 2016), and extract the 2048-dimensional output of the final hidden layer, to be treated as the representation of that image. We use these data to train several different configurations for the VAE. Our VAE consists of a feed-forward encoder and decoder network, each with two dense hidden layers with ReLU activation. We define the hyperparameter d, the dimensionality of the latent space. We experiment with $d \in 2, 3, 4, 8, 16, 32, 64, 128, 256$. Hidden layers are scaled proportionally to the size of the latent space, while input/reconstruction layer sizes are fixed at 2048. We train each of these with an Adam optimizer with a learning rate of 0.001. We save the weights with the best validation loss, stopping training after 5 epochs without improvement. Training takes only a few minutes on a desktop with an Nvidia GTX 1070 GPU.

3.3 Dimensionality Reduction

Due to the exponential complexity of algorithms used to compute convex hulls (specifically Quick-Hull (Barber et al., 1996)) we are unable to compute Classical-based representations for values of $d > 4$. For now, we address this by training the VAE with higher dimensional encodings, then projecting into a lower dimension before applying the Classical-based method. We report results for projected and unprojected variants of both Classical-based and Prototype-based methods in Section 5. Although initial experiments do not suggest a benefit to using higher dimensions (i.e. $d = 4$ dimensions did not outperform $d = 2$ in our early experiments), a priority of our future work is to employ more sophisticated algorithms from computational geometry which will allow us to compute convex hulls in higher-dimensional spaces.

4 Evaluation

4.1 Entailment

For entailment, we consider both the traditional version of the task, in which entailment judgments are binary, as well as a graded variant of the task, in which concepts are said to entail one another to varying degrees (e.g. a *"robin"* is said to be a better instance of *"bird"* than a *"penguin"* is, and thus *"robin"* entails *"bird"* more than *"penguin"* entails *"bird"*). The observation that humans produce graded entailment judgments is what spurred Prototype Theory initially (Rosch and Lloyd, 1978), and thus is a relevant evaluation task. Examples of binary and graded entailment judgements are given in Table 1.

Standard WBLESS		Graded HYPERLEX	
stove→object	✓	kangaroo→animal	6.0
scarf→garment	✓	mammal→animal	6.0
pistol→ weapon	✓	grape→food	5.9
grain→corn	X	animal→mammal	0.8
telephone→stove	X	horn→car	0.9
jacket→raincoat	X	plate→spoon	0.2

Table 1: Positive and negative examples from each of our lexical entailment (LE) evaluation sets.

WBLESS. For the standard (binary) lexical entailment task, we use the WBLESS lexical entailment dataset (Weeds et al., 2014), which consists of 1,168 word pairs, containing an equal number of positive and negative lexical entailment examples. Positive examples are hyponym-hypernym pairs, where negative examples include reversed entailment pairs, co-hyponyms, holonym-meronym pairs, and random word pairs.

HYPERLEX. For the graded entailment task, we use the HYPERLEX (Vulić et al., 2017) dataset, which contains human judgements of the degree of lexical entailment in the range $[0, 6]$. We use the noun component of HYPERLEX, which contains 2,163 noun pairs with a mean score of 3.3.

IMAGENET Mapping. For each word/concept C in WBLESS, we want to obtain a set of images $\mathcal{X}_C$ that are considered instances of that concept. To do this, we compute the hyponym closure of C in WORDNET (containing all hyponym descendants, or all words that entail C), and gather any that exist as IMAGENET class labels. For example, for the WBLESS concept *"bird_of_prey"*, we identify IMAGENET class labels {*"kite"*, *"bald_eagle"*, *"vulture"*, *"great_grey_owl"*}. All image instances in these classes are then considered to comprise $\mathcal{X}_{bird\_of\_prey}$. Often, different concepts map to the same synset. For example, *"toad"*→*"frog"* becomes *"frog"*→*"frog"*. Different pairs also map to identical pairs in IMAGENET. For example, *"lizard"*→*"animal"* and *"lizard"*→*"creature"* each map to *"lizard"*→*"animal"*, despite having different human judgement values. Finally, some pairs might map onto multiple synsets. In the former two cases, we leave these flaw as-is. In the third case, we assign words to their first sense. Experiments with multiple ways of processing these conflicts showed no noticeable impact on results.

After filtering out pairs in which one or both words have no corresponding images in IMAGENET, both of our datasets are left with a slight entailment bias. Specifically, for WBLESS, we are left with 463 examples (325 entailing, 138 non-entailing). For HYPERLEX, we are left with 362 pairs, with a mean score of 4.0.

4.2 Categorization

We frame categorization as a binary classification task for each of the 1000 base-level IMAGENET categories. For each category, we take the 100 positive examples, and 100 random negative examples (from test data). We then evaluate whether each instance belongs to that category.

5 Results

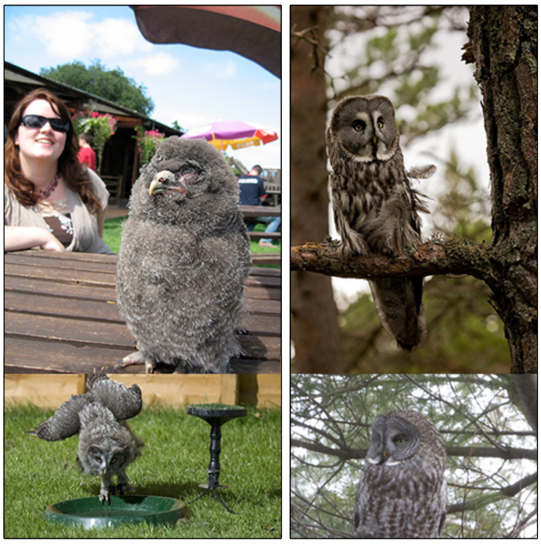
Quantitative results are shown in Table 2. Figure 2 shows illustrative examples of instances occurring near the prototype vs. on the boundary, to provide an intuition of the differences between the two representations.

5.1 Model Variants

We consider several variants of each representation. For the Classical-based representation, we consider both strict and soft variants (Section 3.1). For the Prototype-based method, we train at various dimension sizes and find that $d = 64$ consistently performs best on a held-out validation set. For the Classical-based methods, we find that $d = 2$ consistently performs best on validation. To make as fair a comparison as possible, we also evaluate both methods on representations achieved by training the VAE with $d = 64$ and then projecting down to 2 dimensions. We note that this leads to rough comparisons, and in future work, we intend to find computational approaches which will allow us to compute the Classical-based representations directly in high dimensions.

5.2 Lexical Entailment.

On lexical entailment, the best variant of the Classical-based approach achieves a very high accuracy of 0.90. The method based on a strict interpretation of Classical Theory ($\tau = 1$) achieves a very high precision of 0.99 on WBLESS. While our results are not directly comparable to prior work (since we are using only a subset of WBLESS), we note that this accuracy is quite high for the task. For reference, prior work which used image generality for lexical entailment achieves a maximum accuracy of 0.75 on WBLESS (Kiela et al., 2015a); an approach using hierarchical embeddings achieves an accuracy of 0.87 (Nguyen et al., 2017); and recent work using a retrofitting approach reports an accuracy of 0.91 (Vulić and Mrkšić, 2017). In contrast, the Prototype-based approach greatly over-predicts lexical entailment, yielding high recall and low precision. The two-dimensional and downward-projected configurations perform no better than random, and the 64-dimensional case is only marginally better.



(a) On the boundary (b) Prototypical

Figure 2: Examples instances of *great_grey_owl*. Instances (a) on the Classical-based convex hull boundary are on the left; instances (b) of the most “prototypical” owls are on the right.

We were surprised to find that the Classical-based method also performed better than Prototype-based on graded lexical entailment (HYPERLEX), achieving a Spearman ρ score of 0.55 in both the strict and soft two-dimensional cases. By comparison, Vulić and Mrkšić 2017 achieve a maximum Spearman ρ of 0.71 on HYPERLEX nouns, while work using Poincaré embeddings for learning hierarchical representations achieves a ρ of 0.51 (Nickel and Kiela, 2017). The Prototype-based approach again performs only somewhat better than random on

			Standard LE (WBLESS)				Graded LE (HyperLex)	Categorization (ImageNet)			
Model	Dim.	Proj.	Acc.	Prec.	Rec.	F1	Spearman ρ	Acc.	Prec.	Rec.	F1
Random			0.70	0.70	1.00	0.82	0	0.50	0.50	1.00	0.67
Classical-based (strict)	2	–	0.81	**0.99**	0.72	0.83	0.55	-	-	-	-
Classical-based (soft)	2	–	**0.90**	0.95	0.89	**0.92**	**0.55**	0.55	0.52	1.00	0.69
Classical-based (soft)	64	2	0.87	0.90	0.90	0.90	0.51	0.50	0.50	1.00	0.67
Prototype-based	2	–	0.67	0.67	0.97	0.80	0.08	0.59	0.56	0.9	0.67
Prototype-based	64	2	0.67	0.67	**0.98**	0.80	0.04	0.50	0.50	1.00	0.67
Prototype-based	64	–	0.76	0.76	0.95	0.84	0.20	**0.72**	**0.66**	**0.92**	**0.77**

Table 2: Results comparing Classical-based and Prototype-based approaches on lexical entailment (WBLESS and HYPERLEX) and categorization (IMAGENET).

HYPERLEX, with the 64-dimensional configuration performing best. We were surprised to find that the Prototype-based method performed worse on graded entailment, since Prototype Theory should be well-suited to capturing graded judgements. Further experiments are required to diagnose the extent to which the poor performance of the Prototype-based methods on lexical entailment are due to theory vs. in particulars of our instantiation.

Categorization. The only approach that performs significantly better than random on categorization is 64-dimensional Prototype-based. All 2-dimensional cases (real and projected) perform at chance, over-predicting positive categorizations. This is unsurprising, as it can be expected that more dimensions are needed to capture sufficient information for differentiating classes. We note that, since our image instances are represented as pretrained IMAGENET classifier embeddings, high categorization accuracy can be achieved with a simple perceptron. However, we are not interested in the task of categorization *per se*. Rather, our goal is to assess the extent to which a single representation of a concept can be used to perform both categorization and entailment, without training task-specific modules.

6 Discussion

Several aspects of these initial results prevent us from drawing strong conclusions. In particular: the fact that we cannot compare the representations directly in high dimensions, the fact that we focus on a small number of concrete nouns only, and the fact that we choose one particular definition of prototype and distance function despite the existence of many equally-plausible alternatives.

Nonetheless, despite being preliminary, our results suggest trends which are intuitive as well as some which are counter-intuitive. In particular, we were unsurprised to find that Classical-based representations achieve high precision and all-around high accuracy for tasks related to entailment. As this theory was largely developed with the goal of explaining logical inferences, it is intuitive that such representations would be more sensitive to distinctions which explain judgements about entailment. Similarly, we were unsurprised to see that the Prototype-based representations achieve better performance at categorization, as such theories were originally motivated in terms of categorization (rather than inference) phenomena.

The strong performance of the Classical-based method on the graded entailment evaluation was highly unexpected. Further investigation is required in order to understand whether these results are attributable to something superficial (e.g. artifacts of the dataset), something methodological (e.g. our choice of distance function), or something deeper about the relationship between these two theories. However, this counter-intuitive result does emphasize how aspects of Classical Theory (i.e. the explicit representation of a "boundary") can play a role in the representation of concepts without sacrificing the ability to make graded or probabilistic predictions.

7 Related Work

Our work is very closely related to the work of Young et al. (2014), which sought to instantiate the formal semantics notion of set-theoretic entailment using images to represent the "worlds" to which natural language refers. Their work focused on representations motivated by Classical Theory, and dealt with literal sets of discrete images, meaning it could not generalize to referents outside the training data. Our Classical-based method can be viewed as an updated version of their approach, which uses a VAE in order to represent the visual world in a more flexible way. Our Prototype-based method is novel with respect to the work done by Young et al. (2014).

Also very closely related is Kiela et al. (2015b), which represented a lexical concept as a set of image encodings, and sought to make lexical entailment decisions by comparing how dispersed versus compact images within a category were. We note many aspects of Kiela et al. (2015b)'s approach which overlap with our own–namely, the use of sets of images to derive representations of concepts and the use of set overlap to determine entailment. However, our focus is on a particular question which is tangential to Kiela et al. (2015b). That is, we are interested in the differences between boundary-focused (Classical) representations compared to center-focused (Prototype) representations, acknowledging either representation is equally capable of capturing properties like dispersion and "generality" of a concept, the focus of Kiela et al. (2015b)'s work.

In general, the present study relates to the ample prior work on visually-grounded meaning representations. Beinborn et al. (2018) gives a in-depth survey of work in this area, from both a computational and a cognitive perspective. Of particular relevance to our work is prior work on multimodal lexical semantics, e.g. work which extends skipgram-like training procedures to include both visual and text information Lazaridou et al. (2015); Silberer and Lapata (2012); Silberer et al. (2017); Collell et al. (2017); Kiela et al. (2016); Kiros et al. (2018). Such representations not only perform better in practice, but have been shown to be more cognitively-plausible in terms of their ability to predict human brain activity (Bulat et al., 2017). Beyond lexical representations, multimodal representations have been incorporated representations of more complex concepts such as frames (Shutova et al., 2017) and full sentences (Han et al., 2017). Again, our work differs in that we are not focused on harnessing visual data *per se*; rather, our focus is on how, given a representation of the world to which we can "ground" meaning, different theories can be operationalized, and how the assumptions of these theories affect performance on basic tasks. That is, we view our work as complementary to, rather than competing with, existing ongoing work on multimodal and grounded representations.

Finally, there is an enormous body of work aimed at modelling lexical entailment using text-only training data, recently (Shwartz et al., 2016; Chang et al., 2017; Vulić and Mrkšić, 2017; Pavlick and Pasca, 2017; Pavlick et al., 2015). Such work often treats lexical entailment as a supervised learning problem, or at least as a task to which we should tune directly. We view such approaches as fundamentally different from what we present here. That is, our work focuses on how to form concepts which relate language to the world, with the assumption that inferences about entailment should come from reasoning directly about the extensions of these concepts, rather than indirectly by relating the surface forms which refer to those denotations.

8 Conclusion

Using a VAE to encode image embeddings into a shared low-dimensional space, we compare a Classical-based with a Prototype-based model of concepts using common evaluations on lexical entailment and categorization. The Classical-based approach performed exceptionally well on lexical entailment detection, and relatively well on graded entailment judgements. While the higher-dimensional Prototype-based approach performed well on categorization, in general our Prototype-based approach performs subpar. The extent to which this is theory vs. approach can't be determined by this research - the vagueness of the distance function d proposed by Prototype Theory gives way to a vast world of unexplored cognitively plausible instantiations that we look forward to exploring.

References

C Bradford Barber, David P Dobkin, David P Dobkin, and Hannu Huhdanpaa. 1996. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483.

Lisa Beinborn, Teresa Botschen, and Iryna Gurevych. 2018. Multimodal grounding for language processing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2325–2339, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Luana Bulat, Stephen Clark, and Ekaterina Shutova.

2017. Speaking, seeing, understanding: Correlating semantic models with conceptual representation in the brain. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1081–1091, Copenhagen, Denmark. Association for Computational Linguistics.

Haw-Shiuan Chang, Ziyun Wang, Luke Vilnis, and Andrew McCallum. 2017. Distributional inclusion vector embedding for unsupervised hypernymy detection.

Guillem Collell, Ted Zhang, and Marie-Francine Moens. 2017. Imagined visual representations as multimodal embeddings. In *AAAI*, pages 4378–4384.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee.

Dan Han, Pascual Martínez-Gómez, and Koji Mineshima. 2017. Visual denotations for recognizing textual entailment. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2843–2849.

Hans Kamp and Barbara Partee. 1995. Prototype theory and compositionality. *Cognition*, 57(2):129–191.

Douwe Kiela, Laura Rimell, Ivan Vulić, and Stephen Clark. 2015a. Exploiting image generality for lexical entailment detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 119–124.

Douwe Kiela, Laura Rimell, Ivan Vulić, and Stephen Clark. 2015b. Exploiting image generality for lexical entailment detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 119–124, Beijing, China. Association for Computational Linguistics.

Douwe Kiela, Anita Lilla Ver, and Stephen Clark. 2016. Comparing data sources and architectures for deep visual representation learning in semantics. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 447–456, Austin, Texas. Association for Computational Linguistics.

Diederik P Kingma and Max Welling. 2013. Autoencoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Jamie Kiros, William Chan, and Geoffrey Hinton. 2018. Illustrative language understanding: Large-scale visual grounding with image search. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 922–933, Melbourne, Australia. Association for Computational Linguistics.

Stephen Laurence and Eric Margolis. 1999. Concepts and cognitive science. *Concepts: core readings*, pages 3–81.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. pages 153–163.

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. 1990. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244.

Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Hierarchical embeddings for hypernymy detection and directionality. *arXiv preprint arXiv:1707.07273*.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in neural information processing systems*, pages 6338–6347.

Daniel N Osherson and Edward E Smith. 1981. On the adequacy of prototype theory as a theory of concepts. *Cognition*, 9(1):35–58.

Ellie Pavlick, Johan Bos, Malvina Nissim, Charley Beller, Benjamin Van Durme, and Chris Callison-Burch. 2015. Adding semantics to data-driven paraphrasing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1512–1522, Beijing, China. Association for Computational Linguistics.

Ellie Pavlick and Marius Pasca. 2017. Identifying 1950s american jazz musicians: Fine-grained isa extraction via modifier composition. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2099–2109, Vancouver, Canada. Association for Computational Linguistics.

Eleanor Rosch and Barbara Bloom Lloyd. 1978. Cognition and categorization.

Ekaterina Shutova, Andreas Wundsam, and Helen Yannakoudakis. 2017. Semantic frames and visual scenes: Learning semantic role inventories from image and video descriptions. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 149–154, Vancouver, Canada. Association for Computational Linguistics.

Vered Shwartz, Enrico Santus, and Dominik Schlechtweg. 2016. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection.

Carina Silberer, Vittorio Ferrari, and Mirella Lapata. 2017. Visually grounded meaning representations. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2284–2297.

Carina Silberer and Mirella Lapata. 2012. Grounded models of semantic representation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1423–1433, Jeju Island, Korea. Association for Computational Linguistics.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Ivan Vulić, Daniela Gerz, Douwe Kiela, Felix Hill, and Anna Korhonen. 2017. Hyperlex: A large-scale evaluation of graded lexical entailment. *Computational Linguistics*, 43(4):781–835.

Ivan Vulić and Nikola Mrkšić. 2017. Specialising word vectors for lexical entailment.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259. Dublin City University and Association for Computational Linguistics.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. volume 2, pages 67–78.

Lotfi Asker Zadeh, George J Klir, and Bo Yuan. 1996. *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers*, volume 6. World Scientific.