Encoding Lexico-Semantic Knowledge using Ensembles of Feature Maps from Deep Convolutional Neural Networks

Steven Derby **Paul Miller** **Barry Devereux**

Queen's University Belfast, Belfast, United Kingdom
{sderby02, p.miller, b.devereux}@qub.ac.uk

Abstract

Semantic models derived from visual information have helped to overcome some of the limitations of solely text-based distributional semantic models. Researchers have demonstrated that text and image-based representations encode complementary semantic information, which when combined provide a more complete representation of word meaning, in particular when compared with data on human conceptual knowledge. In this work, we reveal that these vision-based representations, whilst quite effective, do not make use of all the semantic information available in the neural network that could be used to inform vector-based models of semantic representation. Instead, we build image-based meta-embeddings from computer vision models, which can incorporate information from all layers of the network, and show that they encode a richer set of semantic attributes and yield a more complete representation of human conceptual knowledge.

1 Introduction

Many approaches to representing the meaning of imageable, concrete concepts (e.g. FROG, APPLE, CAR, GUITAR) have been developed in the fields of cognitive science, computational linguistics and computer vision. Most explicitly, property listing studies have been used in cognitive psychology and cognitive neuroscience to characterise word meaning in terms of discrete semantic properties (McRae et al., 2005; Devereux et al., 2014; Buchanan et al., 2019). In property listing studies, human participants enumerate as many features as they can for each concept word, and these responses are then aggregated and normalised to a set of verbal semantic descriptors that correspond to elements of concept meaning (e.g. *does-croak* for FROG). This gives a representation of each concept as a sparse vector which encodes the semantic properties that occur for that concept. The resulting properties can then been applied to research investigating the organisation of semantic processing across the cortex, and to studies of the speed or ease of semantic processing for different concepts and different types of concept knowledge (Fieder et al., 2019; Evans et al., 2019; Kivisaari et al., 2019a; Bruffaerts et al., 2019).

A desirable trait of semantic property norms is their interpretability, since this interpretability facilitates the design of cognitive experiments on conceptual semantics (Murphy, 2004). This interpretability has also allowed researchers in NLP interested in distributional lexical semantics to gain better insights into the kinds of information that dense vector space models attain from pretraining. Even though many state-of-the-art vector space models of word meaning perform well when evaluated on both intrinsic and downstream tasks (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2017), researchers have demonstrated that such models often fail to fully encode certain facets of conceptual meaning (Li and Gauthier, 2017). For example, taxonomic properties (i.e. properties describing the object category, such as *is-an-amphibian* for FROG) and properties reflecting encyclopedic information or information about object function tend to be well-represented in the vector space, but properties that correspond to other kinds of attributes, such as colour, form, and modes of motion, tend to be poorly encoded (Rubinstein et al., 2015; Collell and Moens, 2016). These insights have motivated the development of

Proceedings of the 28th International Conference on Computational Linguistics, pages 1906–1921
Barcelona, Spain (Online), December 8-13, 2020

multimodal semantic representations, which learn from multiple sources of information to ground these representations in the real world – an approach most successfully demonstrated by models that combine data from text and images (Bruni et al., 2014; Kiela and Bottou, 2014; Lazaridou et al., 2015b; Silberer, 2017).

Generally, when image data is incorporated into the construction of word embedding spaces, vectors are derived from the penultimate layer of a deep convolutional neural network trained on an image classification task. Although not usually explicitly stated, the rationale for using the penultimate layer of the network is that this layer should be the layer that is maximally relevant to the predicted label for the objects, whilst also being the layer that is least influenced by the low-level visual noise due to the details of the sampled training images. However, it is possible that meaningful object knowledge is present in lower layers of the network even if such information is not directly relevant to the object labelling task that the computer vision model is trained on. For example, in human property listing data, the property *is-green* is reliably given for FROG. In a computer vision model trained to discriminate between object classes, however, the greenness of frogs may not be strongly relevant to discriminating between images of frogs and images of other amphibians or reptiles (which also tend to be green) and so this information – highly relevant to human semantic representations – may not be strongly represented in the penultimate layer.

In this work, we demonstrate that whilst using the penultimate layer of a computer vision model is an effective way to capture concept semantics, this approach is not optimal with respect to the goal of producing representations that encode information about cognitively-relevant semantic attributes of concepts. Using human property norm data, we demonstrate that certain features and feature types are more decodable at particular layers of the network than others. To produce representations that can make use of all the available semantics-relevant information from the network, we merge the feature map information by constructing *meta-embeddings* using information from all convolutional layers. We demonstrate that these ensembles of distributional models produce more complete representations of conceptual meaning, when evaluated against human conceptual knowledge. To our knowledge, ours is the first work to consider and evaluate the use of all layers of computer vision models for constructing semantic models, and is the first to consider a range of recent computer architectures in building ensembled and multi-modal representations. Finally, we demonstrate how these *meta-embeddings* can be used in a zero-shot property mapping task, which allows us to automate the generation of interpretable semantic properties for unseen concepts. We make our code, embeddings, and analysis pipleine openly available[1].

2 Related Work

Whilst word embedding vectors derived from text data have a long history of proven utility on a wide range of downstream tasks, they have been shown to struggle with encoding certain types of semantically-relevant information. Directly analysing the relationship between explicit property knowledge found in property norm data and the information in pretrained distributional models shows that particular properties relating to more sensory or perpetual information about object semantics can be poorly captured, compared with more associative and encyclopedic knowledge (Rubinstein et al., 2015; Collell and Moens, 2016; Li and Gauthier, 2017). Sommerauer and Fokkens (2018)), using probing classifiers, showed that many of these properties may not be decodable at all from text-based embedding spaces. Findings such as these have motivated researchers to incorporate multimodal information into representations of word meaning (Bulat et al., 2016).

Meta-embeddings have emerged as a useful method for combining information from different word embeddings models (Yin and Schütze, 2016). Different embeddings may be trained on various corpora of text, with different sizes, vocabularies, learning methods or model architecture. Meta-embeddings can then be created that combine the complementary information from all sources (Muromägi et al., 2017). In the context of language modelling, different layers of a pre-trained language model may be sensitive to different kinds of linguistic information, and effectively combining embedding information across layers has been shown to improve performance on different tasks sensitive to different kinds of

[1]Github link to code at `https://github.com/stevend94/Decoding-Semantic-Properties`.

information (such as POS-tagging and word sense disambiguation) (Peters et al., 2018). A number of successful methods have emerged for combining word embedding spaces from different sources. Coates and Bollegala (2018) demonstrate that combining vectors using element-wise addition can be just as effective as concatenating, given that the embeddings are orthogonal. Bollegala and Bao (2018) propose a number of autoencoder type networks to combine one or more vector space models. Finally, Neill and Bollegala (2018) give a comprehensive set of empirical results for a number of models and loss functions for learning complex meta-embeddings, demonstrating that loss functions that focus on vector direction such as cosine or KL-divergence based losses give the best performance on intrinsic benchmarks.

In distributional semantics, powerful approaches have been developed for building functions that can map from one semantic space to another (Lazaridou et al., 2014), compelling researchers to construct cross-modal mappings from dense distributional models with much larger vocabularies onto property norm data (Fagarasan et al., 2015). For example, Derby et al. (2019) constructed distributed semantic representations for each property dimension used in a large set of property norms, whilst Li and Summers-Stay (2019) showed that deep neural networks provide the best performance for zero-shot mapping between semantic spaces.

Motivating our work on ensembling over layers of deep convolutional neural network vision models, it has been shown that different layers of such networks learn features that reflect different kinds of visual properties. The lower layers of vision networks tend to capture visually basic features such as Gabor filters and colour gradients, which are then combined at later layers to construct task-specific high-level visual features that are relevant to object classification (Yosinski et al., 2014; Zeiler and Fergus, 2014).

3 Approach

The most prominent method for building image-based semantic models involves using pretrained deep convolutional neural networks (DCNNs) to extract visual information from image data by retrieving the output vectors from the penultimate layer of the network. In this work, we utilise DCNNs trained for the ImageNet LSVRC competition (Deng et al., 2009), which aim to predict the correct object in an image from a set of 1000 possible labels. In general, these networks use convolutional layers to extract visual information and build increasingly complex features, before one or many fully-connected layers are used to compute the probability distribution over the classes. Distributional semantic spaces can then be constructed from the penultimate fully-connected layer. Here, we instead focus our analysis on representational spaces generated at *all* convolutional layers of the network. Our goal is to demonstrate that the types of semantic attributes encoded in these feature maps depend on their depth in the network, and thus using only the penultimate layer may be suboptimal for representing concept meaning.

3.1 Visual Stimulus

We make use of the CSLB property norm data (Devereux et al., 2014), which includes 638 concept words together with 2725 human-elicited semantic properties[2]. We used a script to web scrape ten representative images for each of these concepts from a Google image search. We manually reviewed the images to check that they were appropriate and representative (for example, to ensure images for the search term APPLE do not include the logo of Apple Inc.). For each word, we feed the corresponding images into a DCNN and extract the feature map outputs at every layer of the network. Once we have retrieved the feature maps for each concept, we perform additional preprocessing steps in order to create embedding vectors at each layer. Each convolutional filter should activate if it receives certain visual patterns from the stimulus, with the resulting feature map representing the activity value of each filter at each spatial location. To obtain an overall measure of the presence of each feature in each image, we perform global max pooling across the feature maps, which takes the highest activity value at all spacial locations for each filter. We then average the max-pooled responses for each filter across each of our ten images to get the final concept representation and finally normalise the vectors using $L2$ distance.

[2] https://csl.psychol.cam.ac.uk/propertynorms/

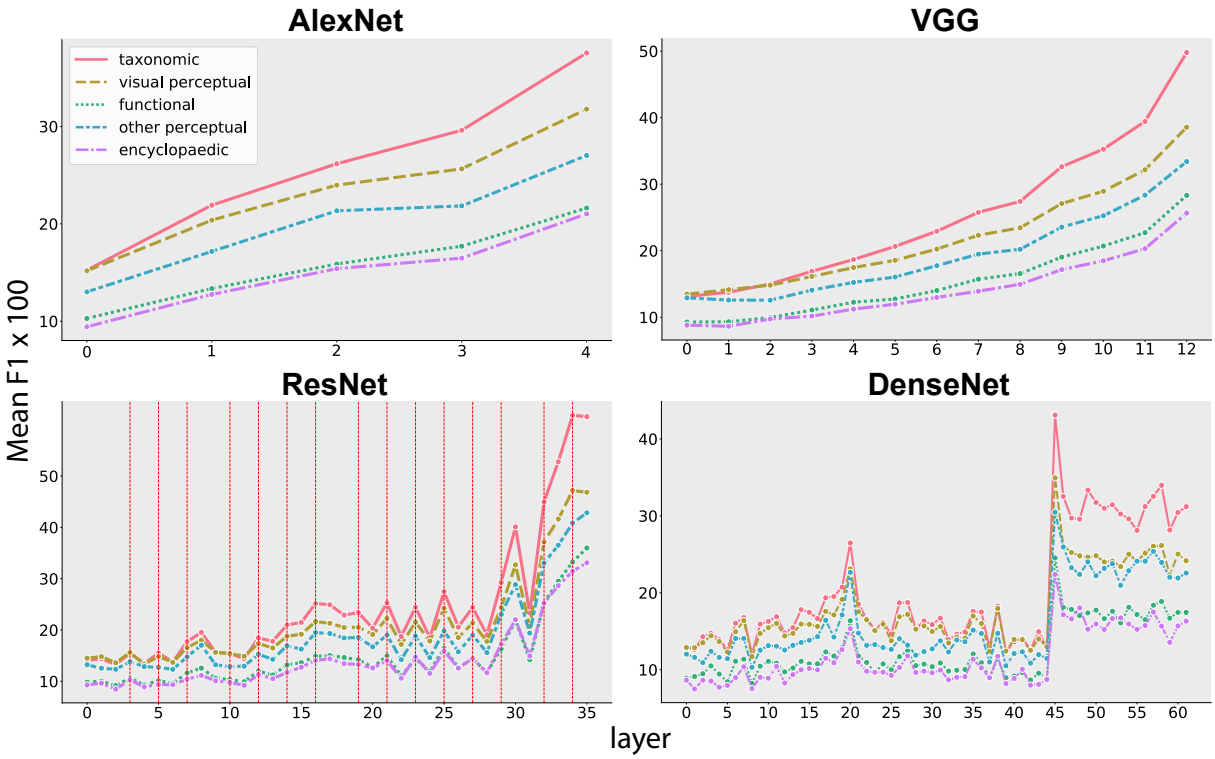


Figure 1: Decodability scores on human property norm data using the convolutional layers of four deep neural networks for image classification (*AlexNet*, *VGG ResNet*, and *DenseNet*. The *ResNet* panel includes red dotted vertical lines indicating the residual connections of the network.

3.2 Image Classification Models

For our analysis, we evaluate a number of standard implementations of networks trained for image classification[3]. The first model is ***AlexNet***, containing five convolutional layers followed by three feed-forward layers. AlexNet is the most widely used pretrained DCNN in work on multimodal distributional semantics. Next, we chose the ***VGG16*** model, which has a similar architecture to *AlexNet* but contains 16 convolutional layers (Simonyan and Zisserman, 2014). By comparing *Alexnet* and *VGG16*, we aim to investigate how depth affects the models' ability to encode human-relevant semantic property knowledge. For our third model, we chose ***ResNet34*** which is not only deeper, with 34 convolutional layers, but also has residual connections between layers. These connections allow information from lower layers to more easily flow to higher layers.The final model we consider is ***DenseNet***; it further extends the objective of feeding low-level information as inputs to the latter layers of the network by using dense skip connections (Huang et al., 2017). As *DenseNet* has a vast number of layers (169), we only take the output of the first convolutional layer, each block layer in the network, and each transition block (which perform downsampling of the feature maps). ResNet and DenseNet have been shown to be amongst the most "'brainlike" DCNNs, insofar as their internal representations correlate well with neuroimaging data from the human visual processing stream (José Meijer and Visser, 2019; Wen et al., 2018).

4 Analysis

To measure how well each layer of a DCNN encodes salient properties of human conceptual knowledge, we train supervised models to predict the presence of a property for a given concept. We note that while a supervised classifier's ability to identify the presence of a property indicates that the property is encoded in the representations, the converse is not always true (Collell and Moens, 2016).

[3] https://pytorch.org/docs/stable/torchvision/models.html

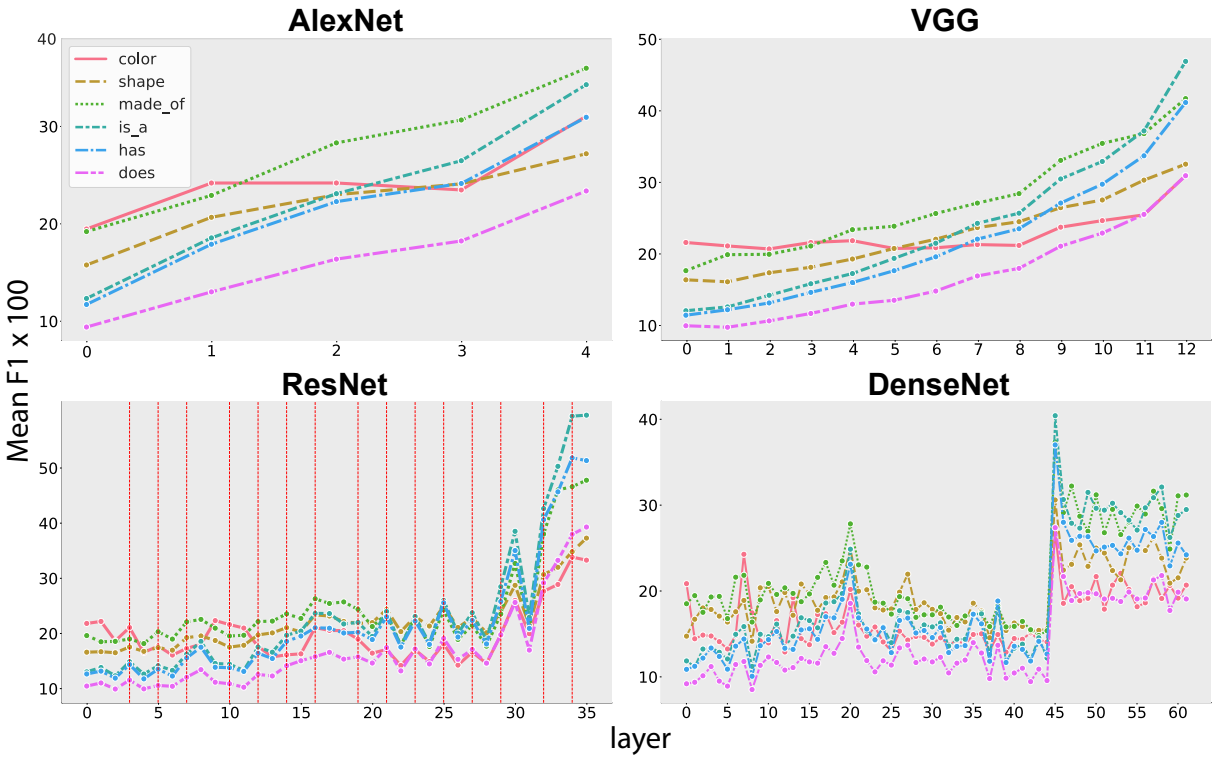


Figure 2: Decodability scores on fine-grained feature categories using the convolutional layers of the four DCNNs. The *ResNet* panel includes red dotted lines indicating residual connections.

4.1 Decoding Property Norms Data from DCNN layers

We first preprocessed the property norm data so as to retain only semantic properties that occur for at least 5 concepts (leaving 638 concepts and 390 properties). For each property, we perform 5-fold cross-validation with stratified sampling so that at least one positive case occurs in each test fold, and train a logistic regression classifier to predict the presence of that property for each embedding, recording the average F1 score over all the folds. Since the dataset is highly imbalanced, we down-weight the negative classes in the loss function and regularise by adding the $L2$ norm of the weights. After obtaining the classification result for each property, we use the partition of properties into distinct feature classes in the CSLB data to aggreggate the results by property class. These property classes are *Taxonomic* (e.g. *is-an-amphibian*), *Encyclopaedic* (e.g. *lays-spawn*), *Functional* (e.g. *hops*), *Visual Perceptual* (e.g. *is-green*) and *Other perceptual* (e.g. *does-croak*).

The results are consistent across all models, with *Taxonomic* being the most decodable followed by *Visual Perceptual*, *Other Perceptual*, *Functional* and finally *Encyclopaedic* (Fig. 1). These results follow previous work where taxonomic features tend to be more decodable than other attributes, for many vector space models. Furthermore, all types of concept properties seem to be more decodable as we move to later layers of the network. Such results could be down to the supervised classifier being unable to make use of the more image-specific information in the lower layers, or the fact that layers in the latter part of the network tend to be better for classification tasks. The most notable exception to this is *DenseNet*, which has two spikes in performance at the convolutional bottleneck layers.

4.2 Fine-Grained Analysis of the Decoding Results

Whilst the five property classes provide a useful distinction between types of semantic information, they tend to include a broad range of semantic attributes. For example, the visual-perceptual class includes visual features relating to colour and texture (*is-green*, *has-smooth-skin*) as well as more complex information about form (*has-four-legs*). To gain a deeper understanding of the kinds of semantic knowledge encoded in the DCNN layers, we divided the properties into more fine-grained classes. First, we split

Model	Encyclo.	Functional	Taxonomic	Visual Perceptual	Other Perceptual	Overall
ResNet	33.09	35.97	61.56	46.84	42.83	44.05
SVD Meta ResNet	32.68	35.92	**68.12**	50.19	48.22	45.80
1ToN Meta ResNet	**34.44**	**37.45**	66.77	**51.28**	**48.25**	**46.74**

Table 1: Average cross-validation F1 scores $\times 100$ for the ResNet embedding space based on the penultimate layer, and the two meta-embedding approaches, for each of the five property classes.

the *Visual Perceptual* features into two basic types of visual information, *Colour* and *Shape/Size*. We expect the lower layers to perform well at decoding these properties since previous research has shown that the lower layers tend to learn colour gradients and Gabor filters (Zeiler and Fergus, 2014). In the CSLB property norm data, semantic properties always consist of a relation term and an attribute value. For example, FROG has the feature *"has-legs"*, where *"has"* is the relation. The property listing task prompted participants to use four such relations: *"is"*, *"has"*, *"does"* and *"made-of"* (Devereux et al., 2014). These relations relate to the type of property being described; for example, *"does"* relates to action or function, while *"has"* corresponds to object parts. We therefore use these four relations to build the other fine-grained categories, for a total of six categories. Based on the previous results, we expect the later DCNN layers to have higher average F1 scores for all properties, but here we are interested in which type of property is most decodable at each layer.

In the lower layers of the DCNNs, we see that *Colour* tends to be the most decodable followed by *"made-of"* and *Shape/Size*, but as we move through to the middle sections of the networks *"made-of"* properties become the most decodable, for all DCNNs (Fig. 2). As we move further, *"has"* and *"is-a"* property decoding improves, and by the end either *"is-a"* or *"made-of"* become the most decodable property types. As we move through the networks, features related to color or shape become relatively less decodable, compared to other feature types. Overall, the results support the rationale that the penultimate layer (as is commonly used) should give a good correspondence to object semantics for the purpose of building distributional semantic models. But would such models also benefit from having direct information about different features from earlier stages of the DCNNs?

5 Improving Distributional Semantic Models with Visual Meta-Embeddings

We have seen that particular layers of DCNNs best capture different types of semantic information. Here we investigate whether we can use this insight to obtain improved image-based semantic embedding spaces and thus build more faithful representations of conceptual meaning. (In these experiments, we focus on *ResNet*, since it gave the best performance on the decoding task; for results with all four models, see the Supplementary Table 1).

5.1 Convolutional Meta-Embeddings

In order to make full use of the information generated by the network, we require a method that can effectively combine the feature maps from each layer, retaining only the most relevant information from each. We construct semantic representations by aggregating features from the output of each convolution layer by assembling them into a single set of representations known as a *meta-embedding*. *Meta-embeddings* are vector representations that incorporate information from a set of word embeddings that can differ in a range of aspects such as training data and training methods (Peters et al., 2018; Coates and Bollegala, 2018). Most importantly, they look to combine complementary knowledge from each embedding, and do not require that the vectors be the same dimensionality. Here we apply two common approaches. The first approach is a simple concatenation technique to combine all embeddings. Following previous work, we also up-weight the best embeddings; in this case, we multiply the second-to-last layer by 5 and the last layer by 10, keeping the other layers the same before concatenating. To reduce dimensionality, we also apply Single Value Decomposition (SVD) to fix dimensionality to 300 while preserving information from the most important features. We refer to these embeddings as ***SVD Meta ResNet***. The second approach uses a method known as *1ToN* (Yin and Schütze, 2016); this method looks to learn set of meta-

	K Nearest Neighbour (K=5)				Ridge Regression				Neural Network (h=1200)			
Hit@T Accuracy	Top 1	Top 5	Top 10	Top 20	Top 1	Top 5	Top 10	Top 20	Top 1	Top 5	Top 10	Top 20
Unimodal Vector Representations												
ResNet	1.95	29.3	50.0	69.1	5.80	29.1	44.8	61.9	3.23	42.0	60.06	75.5
SVD Meta ResNet	2.04	28.3	50.7	68.6	4.92	43.2	61.0	75.3	2.79	45.1	62.4	76.5
1ToN Meta ResNet	**2.76**	29.3	53.4	71.8	5.92	41.4	58.4	74.6	2.82	44.1	61.4	76.0
GloVe	1.76	27.3	53.8	70.6	7.18	38.9	56.6	71.1	5.14	44.0	62.0	76.7
Multimodal Vector Representations												
ResNet + GloVe	2.38	31.7	59.0	76.7	8.87	43.6	60.6	75.5	**6.74**	50.8	69.5	82.6
SVD Meta ResNet + GloVe	2.32	31.6	58.8	76.4	**8.88**	43.8	60.8	75.6	6.24	51.9	**71.1**	83.0
1ToN Meta ResNet + GloVe	2.60	**32.3**	**60.3**	**78.8**	8.18	**48.1**	**66.1**	**80.6**	5.08	**52.6**	70.9	**83.7**

Table 2: Results for zero-shot cross-modal mapping task using several predictive models. The Hit@K tells us the percentage of test features which appear in the top K neighbours with the ground truth representations.

embeddings using a neural network. For each word, we have a meta-embedding vector, for which the network predicts the associated word embedding for each of our vector space models using several linear layers. A network which combines the information from N embeddings will contain N linear layers which map the meta-embedding into the original constituent word embedding spaces. Suppose we have N distributional models $\{W^1, W^2, \dots W^N\}$, with equal vocabulary V, and vector lengths $(a_i)_{i=1}^N \subset \mathbb{N}$. We define the $1ToN$ neural network with an embeddings matrix $E \in \mathbb{R}^{|V| \times k}$ for some size $k \in \mathbb{R}$, with N linear projections of weights $M_i \in \mathbb{R}^{k \times a_i}$ and corresponding biases b_i, $1 \leq i \leq N$. For each word $w \in V$, let $w^i \in W^i$ be it's associated word vector for each $1 \leq i \leq N$, with meta embeddings $E(w)$. We want to minimize the following loss function, for our neural network parameterized by θ:

$$\mathcal{L}(\theta) = \sum_{j=1}^{N} \beta_j (|\hat{w}^j - w^j|^2 + \lambda |M_j|^2) \tag{1}$$

$$\hat{w}^j = M_j^T E(w) + b_j \tag{2}$$

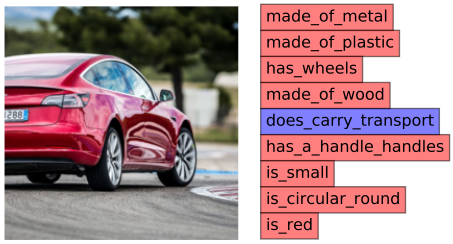
where $[\beta_1, \beta_2 \dots \beta_N]$ are the scaler weightings for each component embedding, though we set these values to one. We instead up-weight the embeddings from the last two convolutional layers which we multiply by 5 and 10 respectively. We call this the ***1ToN Meta ResNet*** embedding.

6 Experiments

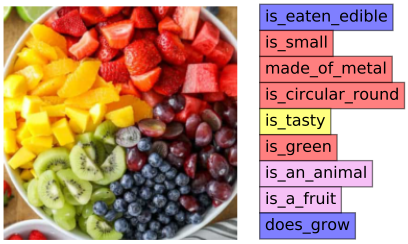
To evaluate these two meta-embedding models, we repeat the decoding experiment with the *SVD* and a *1ToN* meta-embeddings both of size 300 built using the feature maps from all of the *ResNet* convolutional layers. We compare the meta-embeddings with embeddings constructed from the penultimate layer of *ResNet* (i.e. the traditional approach) to see how well they decode each property type.

6.1 Property Decodability

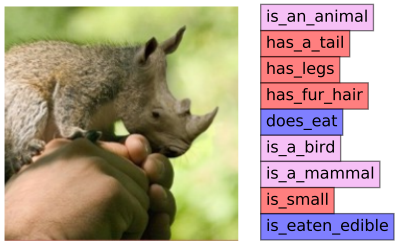
The results are displayed in Table 4. We see that both meta-embedding approaches, ensembling over the convolutional layers of *ResNet*, are better representations for decoding human property knowledge than the traditional approach of using the penultimate layer of *ResNet* alone. We see that there is no real change in how decodable *Encyclopaedic* or *Functional* properties are in the meta embeddings, which is to expected, as this is where text-based word embeddings have been shown to perform strongest. Furthermore, *Taxonomic* and *Visual Perceptual* properties are more decodable since certain layers more strongly encode different types of visual information depending on their location in the DCNN. Surprisingly, *Other Perceptual* information, such as olfactory or taste-based features are also more decodable in the meta-embeddings. Overall, *Taxonomic*, *Visual Perceptual* and *Other Perceptual* have the most significant improvement when using all layers, with the overall F1 score increasing by 5 on average for these three categories, compared with using the penultimate layer of *ResNet* alone.



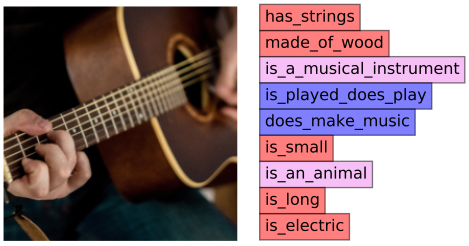
(a) Car - Predictions from *ResNet* convolution **layer 3**



(b) Fruit - Predictions from *ResNet* convolution **layer 15**



(c) Wampimuk - Predictions from *ResNet* convolution **layer 27**



(d) Guitar - Predictions from *ResNet* convolution **layer 35**

Figure 3: Top predictions from the zero-shot cross-modal mapping task for 4 example images. Property predictions for each image are from different convolutional layers of *ResNet*. Each colour represents a feature category: Taxonomic, Visual Perceptual, Functional and Other Perceptual.

6.2 Cross-Modal Embedding-to-Property Mapping

While the sparse property vectors obtained from norming studies are useful in cognitive science due to their interpretability, as a lexical resource they are very limited in size, due to being created manually. This has driven recent work aiming to learn zero-shot cross-modal mappings between a pretrained semantic embedding space and these property vectors, so that property-norm information can be generated automatically from word embeddings (Derby et al., 2019; Fagarasan et al., 2015). Furthermore, we can extend our analysis of the convolutional layers from global to local interpretations based on some sample images (See appendix B). Here we evaluate how accurately our models predict semantic properties for unseen concept words in a zero-shot set-up using several regression models. We predict the property vector for each test concept and find the top T nearest neighbours of the predicted vector, to determine whether the concept word for the ground-truth vector is retrieved within that set, which we refer to as a hit. We perform repeated 10-fold cross-validation on the concepts due to the small number of training samples and average the number of hits across all test folds at each T, for $T \in [1, 5, 10, 20]$ in our evaluations. We perform 5 repeats of each cross-validation. To learn a cross-modal mapping, we report the results for three different models. A k-nearest-neighbours model with $k = 5$, ridge regression and a neural network with one hidden layer. The neural network had a hidden layer of size 1200, $Relu$ activations and used the $Adam$ optimiser. We include a neural network as previous work has shown that they give the strongest performance on this zero-shot cross-modal mapping task (Li and Summers-Stay, 2019). The loss function we use is based on the cosine similarity function from Lazaridou et al. (2014). For each ground truth property norm representation $y \in G$, with corresponding predicted vector $\hat{y}$ from the network parameterised by θ, the loss function is

$$\mathcal{L}(\theta) = \sum_{y \in G} \frac{1}{2}(1 - \cos(y, \hat{y})) \tag{3}$$

Training neural networks on such a small set of data points for zero-shot cross-modal mapping can be difficult, as several problems arise such as *"hubness"* (Radovanović et al., 2010), *"pollution"* (Lazaridou et al., 2015a) and neighbourhood structures resembling the input space more than the output (Collell

Model	MEN (104)	SimLex999 (48)
Unimodal Vector Representations		
ResNet	0.540	0.324
SVD Meta ResNet	0.555	0.360
1ToN Meta ResNet	0.683	0.32
GloVe	0.804	0.210
Multimodal Vector Representations		
ResNet + GloVe	0.829	0.322
SVD Meta ResNet + GloVe	0.829	0.316
1ToN Meta ResNet + GloVe	0.846	0.491

Table 3: Spearman ρ correlation with *MEN* and *SimLex999* human similarity benchmarks.

and Moens, 2018). Hence, we perform a hyperparameter search using a small grid of values with 5-fold cross validation to determine the best set of training parameters. To avoid over-fitting, we determine crossvalidation performance using Mean Average Precision (MAP). To illustrate this cross-modal mappling approach, examples of zero-shot property predictions for held-out images are presented in Figure 3. We also combined the image embeddings with text embeddings to create multimodal distributional models that have been shown to give better performance on cross-modal mapping (Bulat et al., 2016). For the text embeddings, we use *Spacy's GloVe* vectors (Pennington et al., 2014), from the large English language model. To build the text+image multimodal models, we concatenate the $L2$-normalized vectors from the *GloVe* embeddings with each of our image-based embeddings, which gives us three multimodal models in total. The results are presented in Table 2. We see that in all cases, the meta-embeddings outperform the embeddings from the penultimate layer of *ResNet*, and in particular, the $1ToN$ embeddings show the best performance. As the number of models being ensembled increases, information can get lost when concatenating to high dimensionality (Neill and Bollegala, 2018), but with the *1ToN* method the network effectively retains the important information from the ensembled component embeddings because of the learning objective.

6.3 Semantic Similarity Task

A common benchmark to evaluate distributional semantic models is to directly compare word similarity scores with human annotator similarity ratings for word pairs. We utilize *MEN* (Bruni et al., 2012) and *SimLex999* (Hill et al., 2015), for which we have 104 and 48 word pair ratings respectively. In this final evaluation of the models, we use cosine similarity to score word-pair similarity and then use Spearman ρ to measure the correlation between embedding word similarities and the human annotator ratings. As we can see in Table 3, the results again show the same pattern, with the meta embeddings outperforming the penultimate *ResNet* layer for both the unimodal and multimodal (text+image) embeddings.

7 Conclusion

We have demonstrated the potential of utilizing interpretable semantic primitives derived from human property norm data as a tool for investigating the information captured in the latent representations of deep convolutional neural networks. We reveal that, whilst the widely accepted approach for extracting visual semantic representations, using the penultimate layer of DCNNs, yields strong representations of conceptual meaning, they overlook key information generated by the neural network. Instead, we develop meta-embeddings that encompass all the salient feature information encoded in the representations produced at all layers of several DCNNs. These new vector space models are not only closer representations of human conceptual knowledge, but also can be used to build multimodal semantic models that improve performance on a zero-shot cross-modal mapping task and give better fit to human semantic similarity benchmarks. Furthermore, the field of meta-embeddings is rich in potential methods for combining vector space models from different semantic domains, while our research offers empirical evidence that supports our method for constructing meta-embeddings to improve image-based and multi-modal distributional semantic models.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Danushka Bollegala and Cong Bao. 2018. Learning word meta-embeddings by autoencoding. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1650–1661.

Rose Bruffaerts, Simon De Deyne, Karen Meersmans, Antonietta Gabriella Liuzzi, Gert Storms, and Rik Vandenberghe. 2019. Redefining the resolution of semantic knowledge in the brain: advances made by the introduction of models of semantics in neuroimaging. *Neuroscience & Biobehavioral Reviews*.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49:1–47.

Erin M Buchanan, Kathrene D Valentine, and Nicholas P Maxwell. 2019. English semantic feature production norms: An extended database of 4436 concepts. *Behavior research methods*, 51(4):1849–1863.

Luana Bulat, Douwe Kiela, and Stephen Clark. 2016. Vision and feature norms: Improving automatic feature norm learning through cross-modal maps. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 579–588.

Joshua Coates and Danushka Bollegala. 2018. Frustratingly easy meta-embedding–computing meta-embeddings by averaging source word embeddings. *arXiv preprint arXiv:1804.05262*.

Guillem Collell and Marie-Francine Moens. 2016. Is an image worth more than a thousand words? on the fine-grain semantic differences between visual and linguistic representations. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2807–2817.

Guillem Collell and Marie-Francine Moens. 2018. Do neural network cross-modal mappings really bridge modalities? *arXiv preprint arXiv:1805.07616*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Steven Derby, Paul Miller, and Barry Devereux. 2019. Feature2vec: Distributional semantic modelling of human property knowledge. *arXiv preprint arXiv:1908.11439*.

Barry J Devereux, Lorraine K Tyler, Jeroen Geertzen, and Billi Randall. 2014. The Centre for Speech, Language and the Brain (CSLB) concept property norms. *Behavior Research Methods*, 46(4):1119–1127.

Samuel Evans, Cathy J Price, Jörn Diedrichsen, Eva Gutierrez-Sigut, and Mairéad MacSweeney. 2019. Sign and speech share partially overlapping conceptual representations. *Current Biology*, 29(21):3739–3747.

Luana Fagarasan, Eva Maria Vecchi, and Stephen Clark. 2015. From distributional semantics to feature norms: grounding semantic models in human perceptual data. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 52–57.

Nora Fieder, Isabell Wartenburger, and Rasha Abdel Rahman. 2019. A close call: Interference from semantic neighbourhood density and similarity in language production. *Memory & cognition*, 47(1):145–168.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

Anne-Ruth José Meijer and Arnoud Visser. 2019. A shallow residual neural network to predict the visual cortex response. *arXiv*, pages arXiv–1906.

Douwe Kiela and Léon Bottou. 2014. Learning image embeddings using convolutional neural networks for improved multi-modal semantics. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 36–45.

Sasa L Kivisaari, Marijn van Vliet, Annika Hultén, Tiina Lindh-Knuutila, Ali Faisal, and Riitta Salmelin. 2019a. Reconstructing meaning from bits of information. *Nature communications*, 10(1):1–11.

Sasa L Kivisaari, Marijn van Vliet, Annika Hultén, Tiina Lindh-Knuutila, Ali Faisal, and Riitta Salmelin. 2019b. Reconstructing meaning from bits of information. *Nature communications*, 10(1):1–11.

Angeliki Lazaridou, Elia Bruni, and Marco Baroni. 2014. Is this a wampimuk? cross-modal mapping between distributional semantics and the visual world. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1403–1414.

Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. 2015a. Hubness and pollution: Delving into cross-space mapping for zero-shot learning. 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 270–280.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015b. Combining language and vision with a multimodal skip-gram model. *arXiv preprint arXiv:1501.02598*.

Lucy Li and Jon Gauthier. 2017. Are distributional representations ready for the real world? Evaluating word vectors for grounded perceptual meaning. In *Proceedings of the First Workshop on Language Grounding for Robotics*, pages 76–85, Vancouver, Canada, August. Association for Computational Linguistics.

Dandan Li and Douglas Summers-Stay. 2019. Mapping distributional semantics to property norms with deep neural networks. *Big Data and Cognitive Computing*, 3(2):30.

Ken McRae, George S Cree, Mark S Seidenberg, and Chris McNorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37(4):547–559.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Avo Muromägi, Kairit Sirts, and Sven Laur. 2017. Linear ensembles of word embedding models. *arXiv preprint arXiv:1704.01419*.

Gregory Murphy. 2004. *The big book of concepts*. MIT press.

James O' Neill and Danushka Bollegala. 2018. Angular-based word meta-embedding learning. *arXiv preprint arXiv:1808.04334*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Milos Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. 2010. On the existence of obstinate results in vector space models. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193.

Dana Rubinstein, Effi Levi, Roy Schwartz, and Ari Rappoport. 2015. How well do distributional models capture different types of semantic knowledge? 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 726–730.

Carina Silberer. 2017. Grounding the meaning of words with visual attributes. In *Visual Attributes*, pages 331–362. Springer.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Pia Sommerauer and Antske Fokkens. 2018. Firearms and tigers are dangerous, kitchen knives and zebras are not: Testing whether word embeddings can tell. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Brussels, Belgium, November. Association for Computational Linguistics.

Haiguang Wen, Junxing Shi, Wei Chen, and Zhongming Liu. 2018. Deep residual network predicts cortical representation and organization of visual features for rapid categorization. *Scientific reports*, 8(1):1–17.

Wenpeng Yin and Hinrich Schütze. 2016. Learning word meta-embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1351–1360.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.

Appendix A. Supplementary Figure 4.

Model	**Encyclo.**	**Functional**	**Taxonomic**	**Visual Perceptual**	**Other Perceptual**	**Overall**
AlexNet Representations						
AlexNet	20.86	21.91	36.90	32.00	27.07	27.92
SVD Meta AlexNet	27.68	27.81	56.65	42.09	37.62	37.63
1ToN Meta AlexNet	28.40	27.16	54.84	41.67	34.78	37.00
VGG Representations						
VGG	25.90	28.74	49.80	39.29	33.34	35.31
SVD Meta VGG	32.51	34.14	64.27	48.26	42.72	43.71
1ToN Meta VGG	33.16	34.93	64.48	48.71	44.47	44.34
ResNet Representations						
ResNet	33.09	35.97	61.56	46.84	42.83	44.05
SVD Meta ResNet	32.68	35.92	**68.12**	50.19	48.22	45.80
1ToN Meta ResNet	**34.44**	**37.45**	66.77	**51.28**	**48.25**	**46.74**
DenseNet Representations						
DenseNet	16.66	18.02	31.71	24.40	23.07	22.34
SVD Meta DenseNet	28.44	30.35	59.16	43.10	37.88	39.09
1ToN Meta DenseNet	31.10	32.61	59.86	45.64	41.13	41.42

Table 4: Average cross-validation F1 scores $\times 100$ for each model for each of the five property classes. We have included the results of all four DCNN's used in the work.

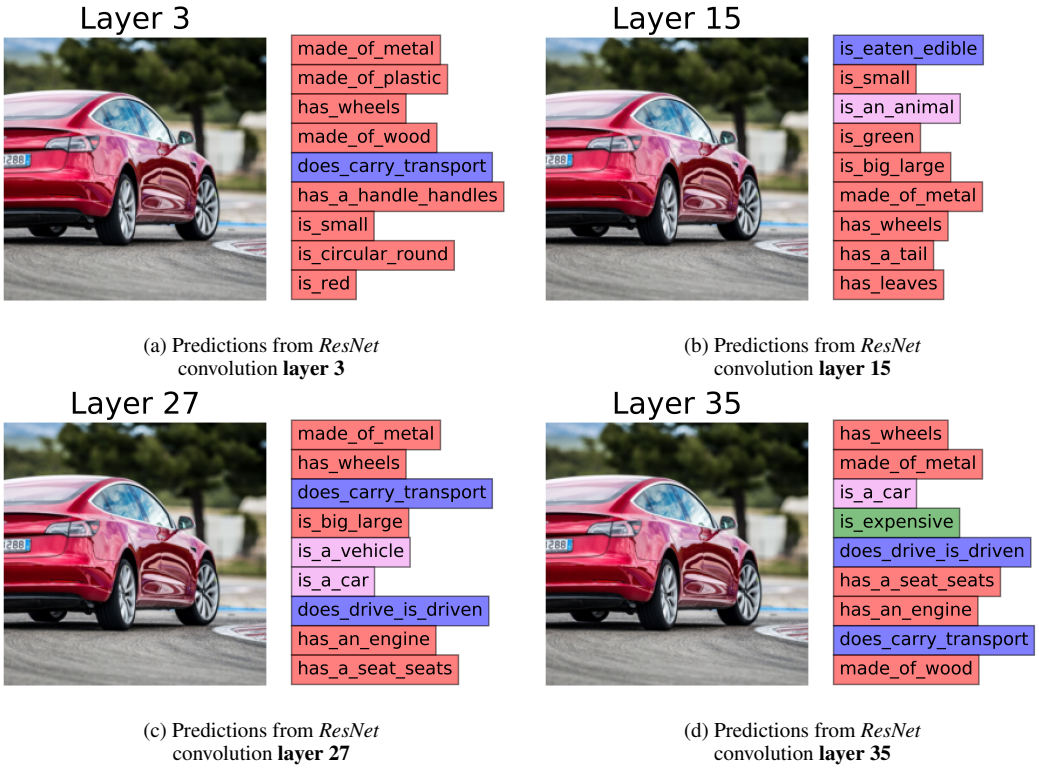
Appendix B. Local Interpretation.

In section 3, we extracted convolutional feature maps from the layers of deep convolutional neural networks for a set of images representing several concepts. We then pool these features based on the concept each image represents, so we could construct semantic representations of word meaning from each convolutional layer of the network. By performing a property decoding task on these embeddings, we could then infer what semantic knowledge the model captures at particular layers of the network. Such an approach reflects a global interpretation of what information the network captures at each convolutional layer, and is not based on any particular sample we gave to the network. Thankfully, cross-modal mapping provides us with a simple method for interpreting local instances from our visual data.

Mapping Images to Semantic Primitives. When we learn a cross-modal mapping from a distributional feature space onto the conceptual feature space, the model must learn to map common features related to a particular concept onto some plausible semantic properties. Because of this, we can use our trained cross-modal map to predict semantic properties for other instances of the concept, since it has been trained to map common feature onto some associated conceptual knowledge. For example, if the model learns to map features it associates as *is-red* based on images from some concept such as ROSE, then a new image of a ROSE should still produce features in the convolutional layers that the cross-modal map similarly identifies as *is-red*. Furthermore, images of other concepts that also have the property *is-red*, such as STRAWBERRY, should be accurately inferred from the model.

For our analysis, we train a ridge regression to map from the convolution layer embeddings of **ResNet** onto the conceptual space. After training, we apply the appropriate image preprocessing to the sample images that we wish to analyse and extract features across all convolutional layers. Since we frame the task as a regression problem, each predicted conceptual mapping should not only predict the correct properties for a concept, but also the strength of the production frequency for each concept. Production frequencies are count-based statistics that reflect the number of times human annotators express that property as true for a particular concept. For our analysis, we predict the conceptual representation for each image and take the highest valued dimensions which correspond to some conceptual property. As there are a large number of layers, we focuses on features at particular intervals of the network, in this

case, layers **3**, **15**, **27** and **35** of **ResNet**.



(a) Predictions from *ResNet* convolution **layer 3**

(b) Predictions from *ResNet* convolution **layer 15**

(c) Predictions from *ResNet* convolution **layer 27**

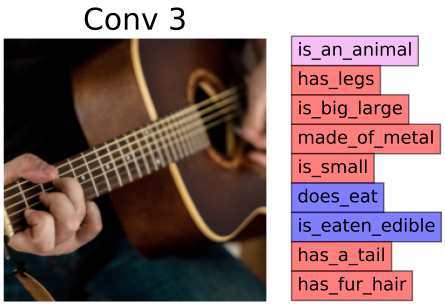
(d) Predictions from *ResNet* convolution **layer 35**

Figure 4: Top property predictions based on the image of a car with features extracted from *ResNet*. Each colour represents a feature category: Taxonomic, Visual Perceptual, Functional and Encyclopaedic.

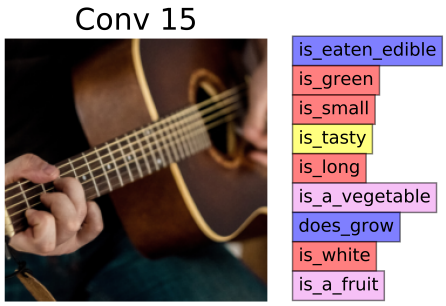
Concept: Car. We first chose an image of the concept CAR (displayed in Figure 4). As we can see the lower layers are dominated by visual perceptual properties, with more high-level properties eventually emerging in the upper layers of the network. Furthermore, the lower layers of the network tend to focus on shape, colour and form across the entire image. We can observe this in the fact that the cross-modal map detects properties such as *made-of-wood*, *is-green* and *has-leaves*, a consequence of the model detecting the trees in the background. Notice also that the dimensions of the object become more precise as we move to the middle layers, with features such as *is-small* and *is-circular-round* appearing at the start, while *'is-big-large* appears in the later layers. Not only do these upper layers predict more complex notions about the concept such as *made-a-metal* or *is-expensive*, but the attention of these features tend to be solely related to the central object in the image, in this case, a car.

Concept: Guitar. Next, we chose an image of the concept GUITAR (displayed in Figure 5), which is another one of the concepts in our lexicon, though is not directly taken from the training data. We see that the model detects some visual properties in the lower layers, which it assumes is related to another high-level concept, an animal. We can see this from the top prediction being *is-an-animal* and other features related to animals such as *has-a-tail* and *has-fur-hair*. It is not surprising that there is a high degree of association between certain properties in the norming study, since many related to particular taxonomies, in this case, *is-an-animal*. Nevertheless, as we move through the network the trajectory of the prediction quickly becomes more related to a guitar, though *is-an-animal* is still predicted in the top ten features. As we can see, these models generalise quite well to other images and can still decode complex features related to conceptual categories.

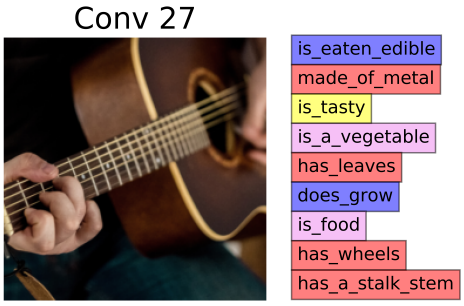
Concept: Fruit. Next, we chose an image of the concept FRUIT (displayed in Figure 6), which is neither taken from the training data or the lexicon, but instead consists of many concepts from the data such as APPLE, BANANA and KIWI. Here we want to analyse how our cross-modal model copes with multiple instances of the concepts. Here, we see that the model can quickly detect visual properties such



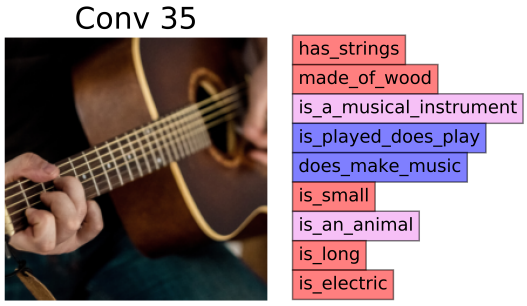
(a) Predictions from *ResNet* convolution **layer 3**



(b) Predictions from *ResNet* convolution **layer 15**



(c) Predictions from *ResNet* convolution **layer 27**

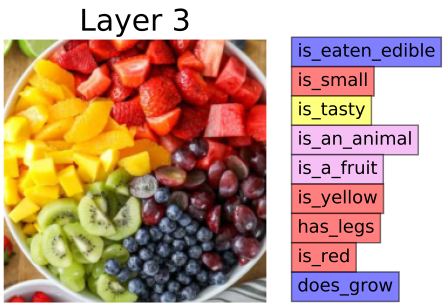


(d) Predictions from *ResNet* convolution **layer 35**

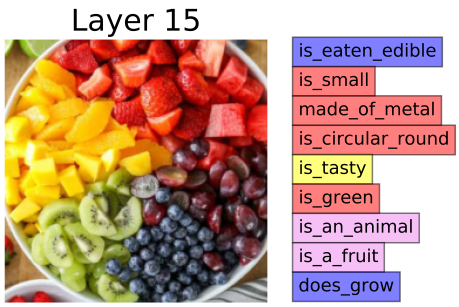
Figure 5: Top property predictions based on the image of a guitar with features extracted from *ResNet*. Each colour represents a feature category: Taxonomic, Visual Perceptual, Functional and Other Perceptual.

as *is-yellow*, *is-red* and *is-small*, though other high-level properties emerge such as *is-eaten-edible* and *is-a-fruit*. Again, *is-an-animal* emerges as a property, which may be due to bias in the model towards high occurring properties.

Concept: Wampimuk. Finally, we chose an imagined concept, known as a WAMPIMUK (displayed in Figure 7). A *Wampimuk* is a fictitious concept proposed by Lazaridou et al. (2014), to convey how context can shape our perception of a concept, even if we have never heard of it before. Humans are capable of building complete semantic representations for concepts, even when the information is fragmented (Kivisaari et al., 2019b). Hence, a sentence like "We found a cute, hairy wampimuk sleeping behind the tree" can communicate a lot of information about what a wampimuk might be, in this case, a small furry animal. The authors create a potential image of such an animal that does not exist, yet we can extract properties about the concept just as well. Hence, we also examine the convolutional layers of the network when given such a creature, to determine whether reasonable semantic properties can be captured by our cross-modal model. We see that the network produces features that the cross-modal map detects as salient aspects of the image such as *is-small* and *has-fur-hair*. Furthermore, the model can detect conceptual knowledge related to this imaginary creature based on the context of this information. For example, in the lower layers, *has-a-tail* is predicted by the cross-modal map, even though there is no evidence of this in the picture, yet it would make sense for a small creature. As we move to the final layer of the network, we can even see complex taxonomies emerge, such as *is-a-mammal* that is quite plausible.



(a) Predictions from *ResNet* convolution **layer 3**

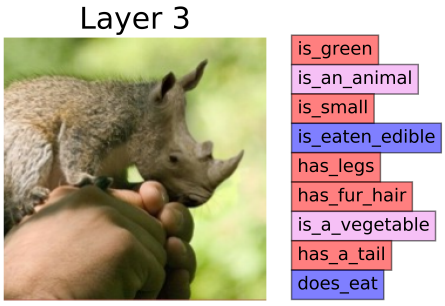


(b) Predictions from *ResNet* convolution **layer 15**

(c) Predictions from *ResNet* convolution **layer 27**

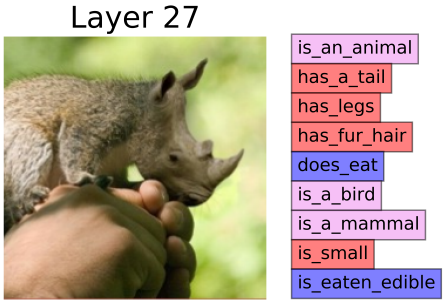
(d) Predictions from *ResNet* convolution **layer 35**

Figure 6: Top property predictions based on the image of fruit with features extracted from *ResNet*. Each colour represents a feature category: Taxonomic, Visual Perceptual, Functional and Other Perceptual.

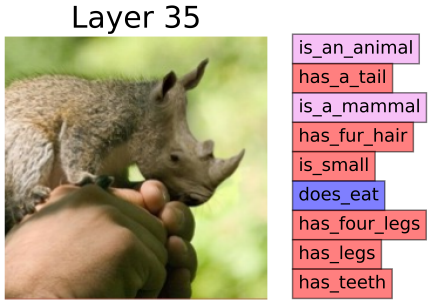


(a) Predictions from *ResNet* convolution **layer 3**

(b) Predictions from *ResNet* convolution **layer 15**



(c) Predictions from *ResNet* convolution **layer 27**



(d) Predictions from *ResNet* convolution **layer 35**

Figure 7: Top property predictions based on the image of a wampimuk with features extracted from *ResNet*. Each colour represents a feature category: Taxonomic, Visual Perceptual, Functional and Other Perceptual.